Green is life
tmg

ABN 43 064 478 842

231 osborne avenue clayton south, vic 3169
PO box 1548, clayton south, vic 3169
t 03 9265 7400 f 03 9558 0875
freecall 1800 680 680
www.tmgtestequipment.com.au

Test & Measurement

- sales
- rentals
- calibration
- repair
- disposal

Complimentary Reference Material

This PDF has been made available as a complimentary service for you to assist in evaluating this model for your testing requirements.

TMG offers a wide range of test equipment solutions, from renting short to long term, buying refurbished and purchasing new. Financing options, such as Financial Rental, and Leasing are also available on application.

TMG will assist if you are unsure whether this model will suit your requirements.

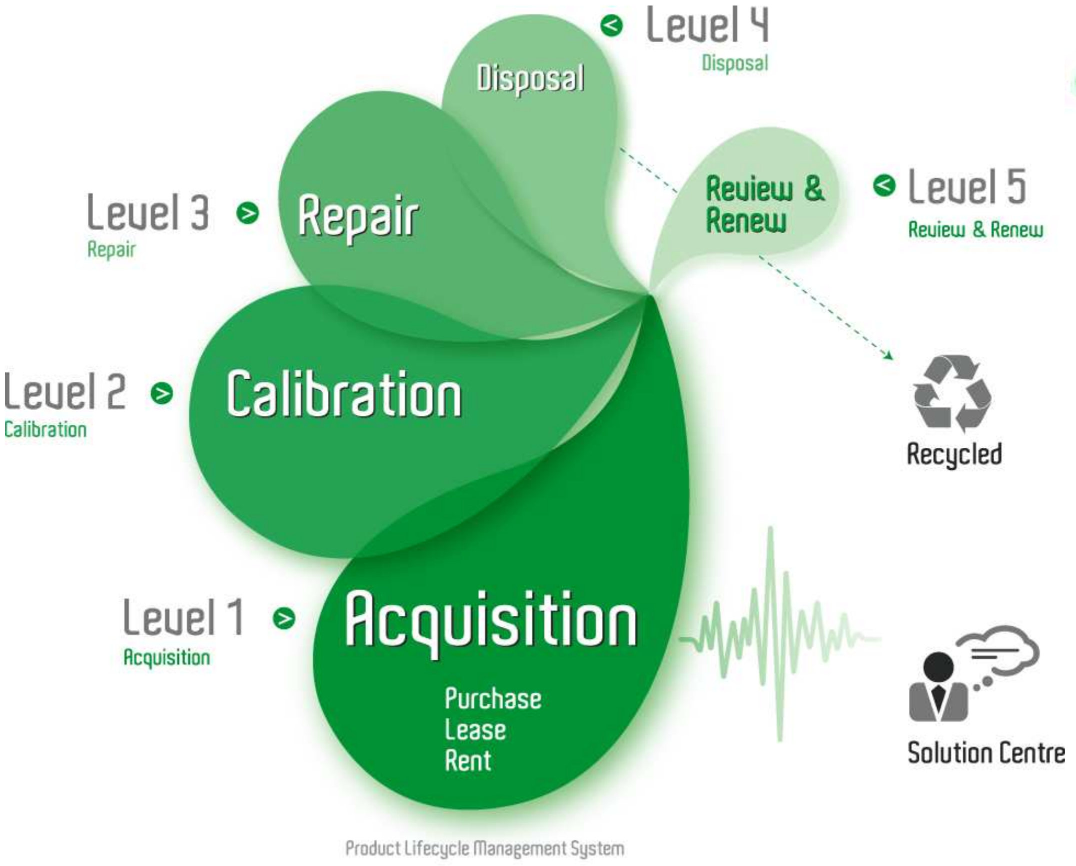
Call TMG if you need to organise repair and/or calibrate your unit.

If you click on the “Click-to-Call” logo below, you can all us for FREE!

TMG Corporate Website

TMG Products Website

Click-to-Call
TMG Now



Disclaimer:
All trademarks appearing within this PDF are trademarks of their respective owners.

User Manual

Tektronix

BPA100
Bluetooth Protocol Analyzer

071-0904-01

This document supports firmware version 2.1 and above.

www.tektronix.com

Tektronix, Inc., P.O. Box 500, Beaverton, OR 97077

WARRANTY

Tektronix warrants that the products that it manufactures and sells will be free from defects in materials and workmanship for a period of three (3) years from the date of shipment. If a product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; c) to repair any damage or malfunction caused by the use of non-Tektronix supplies; or d) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

WARRANTY

Tektronix warrants that the media on which this software product is furnished and the encoding of the programs on the media will be free from defects in materials and workmanship for a period of three (3) months from date of shipment. If a medium or encoding proves defective during the warranty period, Tektronix will provide a replacement in exchange for the defective medium. Except as to the media on which this software product is furnished, this software product is provided "as is" without warranty of any kind, either express or implied. Tektronix does not warrant that the functions contained in this software product will meet Customer's requirements or that the operation of the programs will be uninterrupted or error-free.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period. If Tektronix is unable to provide a replacement that is free from defects in materials and workmanship within a reasonable time thereafter, Customer may terminate the license for this software product and return this software product and any associated materials for credit or refund.

THIS WARRANTY IS GIVEN BY TEKTRONIX IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPLACE DEFECTIVE MEDIA OR REFUND CUSTOMER'S PAYMENT IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

Getting Started

Operating Basics

Reference

Appendices

Glossary

Index

List of Figures

General Safety Summary

Review the following safety precautions to avoid injury and prevent damage to this product or any products connected to it. To avoid potential hazards, use this product only as specified.

Only qualified personnel should perform service procedures.

To Avoid Fire or Personal Injury

Do Not Operate With Suspected Failures. If you suspect there is damage to this product, have it inspected by qualified service personnel.

Do Not Operate in Wet/Damp Conditions.

Do Not Operate in an Explosive Atmosphere.

Keep Product Surfaces Clean and Dry.

Safety Terms and Symbols

Terms in This Manual. These terms may appear in this manual:

⚠ ***WARNING.*** *Warning statements identify conditions or practices that could result in injury or loss of life.*

⚠ ***CAUTION.*** *Caution statements identify conditions or practices that could result in damage to this product or other property.*

Terms on the Product. These terms may appear on the product:

DANGER indicates an injury hazard immediately accessible as you read the marking.

WARNING indicates an injury hazard not immediately accessible as you read the marking.

CAUTION indicates a hazard to property including the product.

Symbols on the Product. These symbols may appear on the product:

⚠	⧈	⏚
CAUTION Refer to Manual	Double Insulated	Protective Ground (Earth) Terminal

Preface

This user manual provides you with the information you need to use the Tektronix BPA100 Bluetooth Protocol Analyzer. The manual is structured as follows:

- *Getting Started* provides a description of the Bluetooth Protocol Analyzer, lists the system requirements, and provides instructions for installing and uninstalling the associated software.
- *Operating Basics* provides instructions for using the Data Collector and the Bluetooth Packet Analyzer.
- *Reference* contains hardware specifications, Bluetooth radio specifications, and sample HCI terminal scripts.
- *Appendix A* contains various regulatory statements.
- *Glossary* contains terms used in the manual.

Reference Documents

The following third-party documents provide additional information:

- *HCI Terminal Guide* (Digianswer #00-11-03) provides information about using a HCI terminal as an interface with Bluetooth hardware.
- *Bluetooth Revealed* (Prentice Hall, Inc., ISBN 0-13-090294-2) provides background on several areas including the basic technology, the Bluetooth specification with information about the protocol stack, Bluetooth profiles, and the future of the technology.
- *Bluetooth: Connect without Cables* (Prentice Hall, Inc., ISBN 0-13-089840-6) provides less background about the technology and more in-depth information about the protocol stack and other areas. This book provides many diagrams.

Contacting Tektronix

Phone	1-800-833-9200*
Address	Tektronix, Inc. 14200 SW Karl Braun Drive P.O. Box 500 Beaverton, OR 97077 USA
Web site	www.tektronix.com
Sales support	1-800-833-9200, select option 1*
Service support	1-800-833-9200, select option 2*
Technical support	Email: techsupport@tektronix.com 1-800-833-9200, select option 3* 1-503-627-2400 6:00 a.m. – 5:00 p.m. Pacific time

* **This phone number is toll free in North America. After office hours, please leave a voice mail message.**
Outside North America, contact a Tektronix sales office or distributor; see the Tektronix web site for a list of offices.

Getting Started

Getting Started

This section contains a description of the Tektronix BPA100 Bluetooth Protocol Analyzer, the system requirements, a list of the product components, and procedures for installing and uninstalling the application software.

Product Overview

The Bluetooth Protocol Analyzer facilitates the development of Bluetooth devices by providing a tool that can nonintrusively and independently intercept the baseband traffic, log, decode, and analyze the packet data transmitted and received over a Bluetooth piconet. The Bluetooth Protocol Analyzer also can function as a prototype debug tool that is capable of participating in a piconet, either as a master or a slave, to initiate various modes of operation, introduce intentional errors, and act as a known reference device.

The Bluetooth Protocol Analyzer consists of a Bluetooth Air Probe with USB connector, a custom USB cable, a CD-ROM containing application software, and a user manual (see Figure 1-1 on page 1-7).

Key Features

The following list notes the key features of the BPA100 Bluetooth Protocol Analyzer:

- Provides decryption in Piconet Mode or Independent Mode. (Version 2.1)
- Enables users to use the HCI terminal application (software provided with Version 2.1) to control the BPA100 in Piconet mode
- Synchronization enhancement provides new capabilities to set drift value in PPM, which is useful when the link goes to sniff, hold, or park mode (Version 2.1)

- Provides capture and display of paging sequence while in Independent Mode and using slave inquiry sync mode (Version 2.1)
- Complies with Bluetooth 1.1 specification (Version 2.1)
- Provides reliable analysis using a fully-compliant product based on proven Digianswer technology
- Operates in either Independent or Piconet (master/slave) mode, which allows you the maximum test and debug flexibility
- Allows you to use advanced triggering and filtering to capture, log, and display only those events or transactions of interest, making it easier to track down faults and optimize storage
- Allows you to use the Free Run Analyzer Display function to continuously monitor the latest session transactions with real-time screen updates while logging directly to the hard disk of the PC
- Provides maximum log history file size by directly logging to the hard drive of the PC, allowing for long-term monitoring of packet traffic to uncover intermittent problems over extended time periods
- Captures and logs all baseband packets transmitted within a Bluetooth piconet, including retransmitted packets, for full session transaction audits
- Isolates, decodes, and displays baseband, LMP, L2CAP, RFCOMM, SDP, OBEX, and TCS commands, events, and data packets for effective visibility into higher protocol layers
- Enhances your control of the application by supporting test modes (in Independent mode, Version 2.1), data whitening, and other low level acquisition parameters

Software and Data Files Included

You are provided with the following software applications and data files on the CD-R that is shipped with the BPA100 Bluetooth Protocol Analyzer:

- Tektronix Bluetooth Data Collector

- Tektronix Bluetooth Packet Analyzer
- Digianswer Bluetooth Neighborhood (version 1.09)
- Digianswer HCI Terminal application
- Samples
- BPA100 User Manual.pdf

Bluetooth Data Collector. You use the Data Collector to set up a log session during which you can intercept all the data transmitted between the devices forming a Bluetooth piconet.

Bluetooth Packet Analyzer. You use the Bluetooth Packet Analyzer to analyze the data logged during a session. The Packet Analyzer can display all the baseband packets logged and isolate, decode, and display LMP, L2CAP, RFCOMM, SDP, OBEX, and TCS packets.

Bluetooth Software Suite. The Bluetooth Software Suite is a collection of Bluetooth applications created by Digianswer. It is composed of the following applications:

- Bluetooth Neighborhood
- Bluetooth Configuration Tool
- Object Editor

Among other functions, you can use the Bluetooth Neighborhood application to do the following:

- Device discovery. Find out which remote Bluetooth devices are available within your range.
- Service discovery. Find out which services (applications) a remote device facilitates.
- Links. Establish links to remote devices.

You can use the Bluetooth Configuration Tool to associate one or more appropriate profiles with a Bluetooth COM port and then add the COM port to your Local Services bar in the Bluetooth Neighborhood window.

You can use the Object Editor to send objects like messages, notes, or business cards if you do not have Microsoft Outlook installed on your system.

For an overview (Beginner's Guide) and detailed information when using the Bluetooth Neighborhood, click the Help button in the application. In addition, three portable document format (PDF) files are installed with the Bluetooth software. These are printable versions of the Help files and the BPA100 manual.

- *Bluetooth Beginner's Guide, An introduction to the Bluetooth Technology*
- *Bluetooth Software Suite User's Manual*
- *BPA100 Bluetooth Protocol Analyzer User Manual*

***NOTE**. While using the Bluetooth Protocol Analyzer, you are advised not to run applications on your computer other than the Packet Analyzer, the Data Collector, and the Bluetooth Neighborhood.*

HCI Terminal. This application allows you to interact with the hardware using an interface similar to the interface provided by an AT Terminal application when communicating with a modem. This facilitates sending HCI commands from the computer to a Bluetooth device and receiving responses. This allows you to test your own Bluetooth hardware. The *HCI Terminal Guide* provides instructions.

***NOTE**. The HCI Terminal and the Bluetooth Neighborhood are different means of creating connections and generating traffic. Only one can be run at a time; you cannot run the HCI Terminal and Bluetooth Neighborhood at the same time.*

Sample Data Files. The Samples folder has log data that you can open and display in the Data Collector and Packet Analyzer without actually having a piconet connection. This folder is not loaded by the installer but can be copied from the CD-ROM.

BPA100 User Manual.pdf. This file is the *BPA100 Bluetooth Protocol Analyzer User Manual* in Portable Document Format. You must use the Adobe Acrobat Reader application to open and print this file. If you do not have a copy of Acrobat Reader, you can download the application from the Adobe web site.

Bluetooth Specification

The Bluetooth Specification is a standard containing the information required to ensure that diverse devices supporting the Bluetooth wireless technology can communicate with each other worldwide. The document is divided into two parts: *Volume 1, Core* and *Volume 2, Profiles*:

- *Volume 1, Core.* This is a lengthy and detailed document that specifies components such as the radio and baseband specifications, link manager protocol, service discovery protocol, transport layer, and interoperability with different communication protocols. It also provides three chapters on test and qualification, including *Bluetooth Test Mode*, *Bluetooth Compliance Requirements*, and *Test Control Interface*.
- *Volume 2, Profiles.* This document specifies the protocols and procedures required for different types of Bluetooth applications, such as service discovery, cordless telephony, serial port, and synchronization profiles.

To access this two-part specification on the Web, go to the following URL address and make your selection:

http://www.bluetooth.com

Bluetooth Protocol Analyzer Configurations

The Bluetooth Protocol Analyzer can be used in two configurations: independent mode or piconet mode.

Independent Mode. Configured as an independent unit, the Bluetooth Protocol Analyzer does not interact directly in the piconet. Instead, after synchronizing to the piconet, it passively monitors the piconet, logging all baseband packets transmitted between the master and the slaves of the piconet. By using advanced triggering and filter features, you can select data of interest to be logged and analyzed after the session is completed. These features are discussed in detail in the *Operating Basics* section.

Piconet Mode. Configured as a participant in the piconet, the Bluetooth Protocol Analyzer uses a fully-protocol stack and participates as the master or a slave in the piconet.

As a master, the Bluetooth Protocol Analyzer logs all baseband packets between itself and the piconet slave device(s). When set up as a slave, it logs all packets between itself and the piconet master device as well as between the master and all other slave devices.

System Requirements

To install and use the application software for the Bluetooth Protocol Analyzer, it is recommended that your system meet the following minimum requirements:

- Computer with a Pentium III (500 MHz or faster); a slower microprocessor can be used but the Data Collector will operate slower when Free Run mode is used
- Microsoft Windows 98, ME, or 2000 operating system
- 128 MB RAM
- Minimum of 200 MB of free space on the hard-disk
- Monitor resolution of 1024 by 768 pixels or higher

Unpacking

The BPA100 Bluetooth Protocol Analyzer package contains the following items (see Figure 1-1):

1. BPA100 Bluetooth Air Probe
2. CD-ROM containing product software
3. *BPA100 Bluetooth Protocol Analyzer User Manual*
4. Custom USB cable

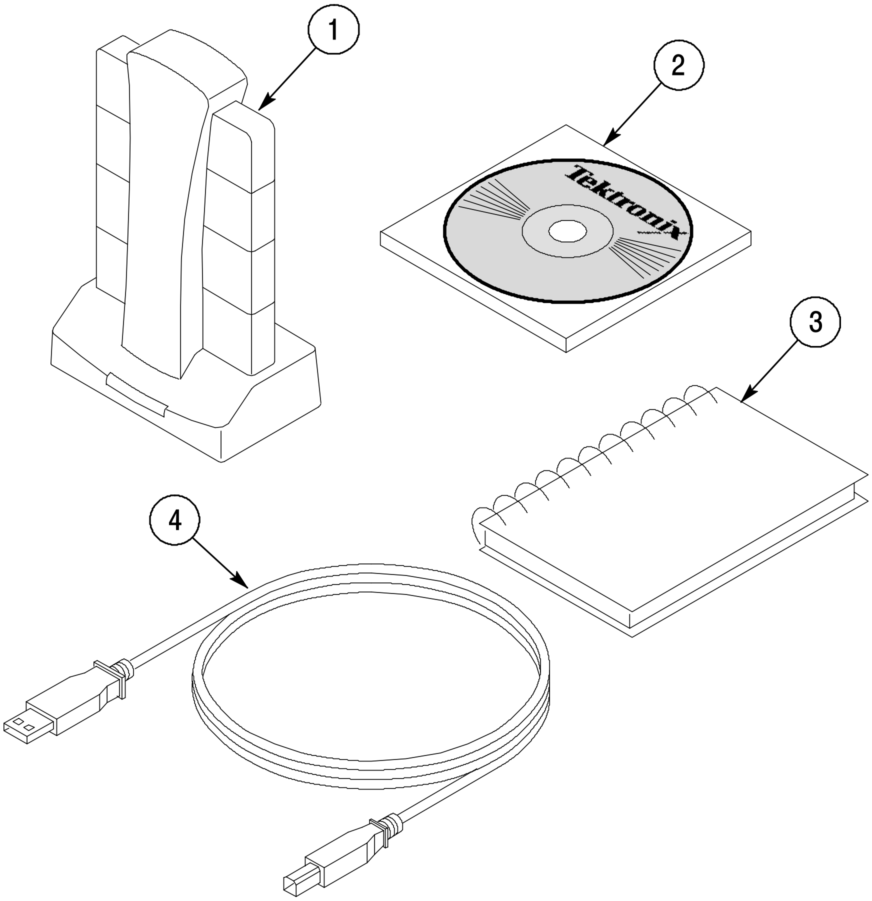


Figure 1-1: Bluetooth Protocol Analyzer components

⚠ ***CAUTION.*** *To ensure compliance with regulatory statements, the custom USB cable included with the BPA100 Bluetooth Protocol Analyzer has additional shielding. Do not use a standard USB cable with this product.*

Replaceable Parts

You can order replacement parts for the following:

- *BPA100 Bluetooth Protocol Analyzer User Manual* (Tektronix replacement part number 071-0904-01)
- Custom USB cable (Tektronix replacement part number 174-4580-00)
- *BPA100 Bluetooth Protocol Analyzer Product Software* (Tektronix replacement part number 063-3469-01)

Installation

The BPA100 installation includes installing hardware, drivers, documentation, and software applications for the Bluetooth Protocol Analyzer. If you are installing this software for the first time, see the *Installation Procedure for New Installs* in this section.

When a new version of the BPA100 software is released, it may necessary to update the firmware resident in the Bluetooth Air Probe as well as the application software, if you are upgrading your software from a previous version. See the *Installation Procedure for Upgrades* in this section.

NOTE. *If a Digianswer Bluetooth DemoCard is installed on your computer, you must uninstall it before you can install the Bluetooth Protocol Analyzer. See* Uninstalling Democard Software on page 1-12. *If you have an older version (v1.0) of the Bluetooth software installed, see* Uninstalling Earlier Bluetooth Software on page 1-12. *For later releases, the installation program uninstalls the the older software for you.*

Installation Procedure for New Installs

1. Insert the Bluetooth Protocol Analyzer CD-ROM.
2. Follow the on-screen steps to complete the installation of the software. Restart your computer when prompted.
3. Connect the USB cable to the Bluetooth Air Probe to an available USB port on the computer.
4. Follow the instructions to install the necessary drivers.

NOTE. *If the hardware requires Windows Ethernet drivers to be installed, you may need your Microsoft Windows installation disk if the necessary files are not located on the hard drive.*

For Windows 2000 installation, the driver installation takes place in several steps, including USB device, Bluetooth USB Device, Bluetooth NAT Protocol, Bluetooth Ethernet Adapter, Bluetooth RFCOMM Protocol, and Bluetooth SDP Protocol.

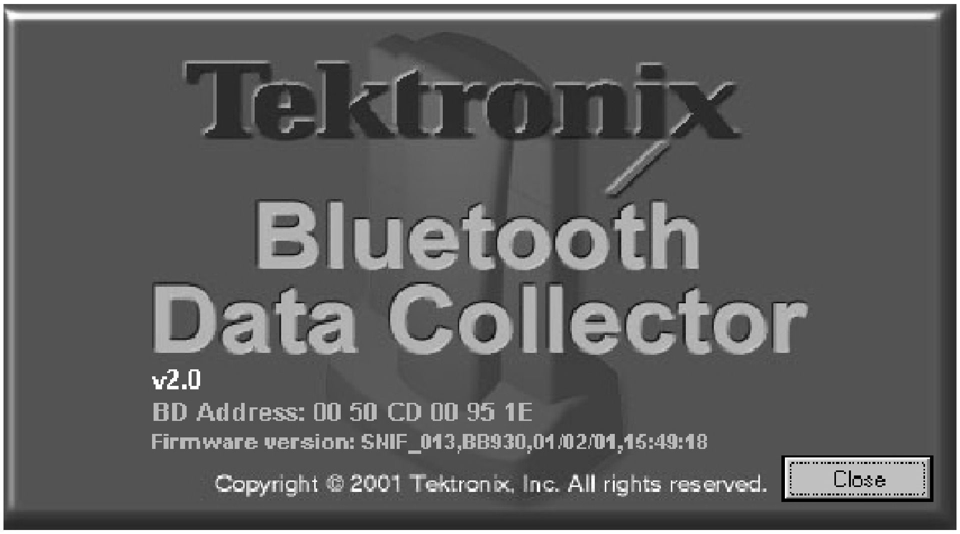
NOTE. *If you must install any drivers manually, they are located on the CD-ROM at D:\Drivers\Win9x for Windows 98 and ME and at D:\Drivers\Win2K for Windows 2000 (where D: is your CD-ROM drive).*

5. Restart your computer. You are now ready to operate your Bluetooth Protocol Analyzer.

***NOTE**. When running Windows 2000, do not disconnect the Bluetooth Probe from the computer unless all the Bluetooth Neighborhood and Bluetooth Data Collector applications are first closed.*

Installation Procedure for Upgrades

1. In the About screen in the Data Collector, note the version of BPA100 software and firmware you are running.



Tektronix

Bluetooth Data Collector

v2.0

BD Address: 00 50 CD 00 95 1E

Firmware version: SNIF_013,BB930,01/02/01,15:49:18

Copyright © 2001 Tektronix, Inc. All rights reserved.

Figure 1-2: Bluetooth Protocol Analyzer About screen

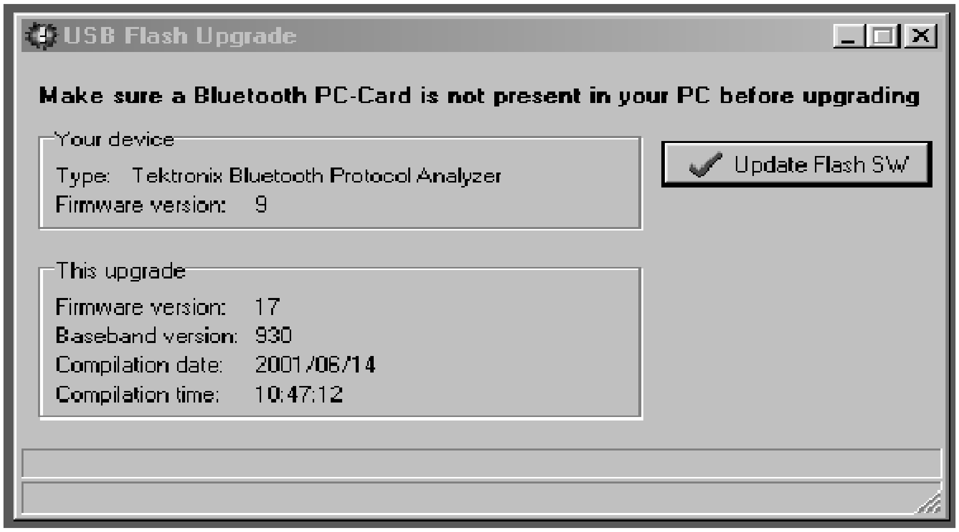
2. In the Help menu of the Data Collector select www.tek.com/bpa_support. This connects you to the BPA100 website. Click on the Software and Drivers link for information on the latest BPA100 software version.

3. If needed, download the latest version from the website.

***NOTE**. A CD copy may also be ordered through your local Tektronix representative.*

4. Unzip the downloaded file into a directory of your choice.

5. Browse to the directory where you unzipped the file, and run the usbflash.exe program. Click the Update Flash SW button and follow the prompts. This upgrades the Bluetooth Airprobe firmware to the latest version.



USB Flash Upgrade

Make sure a Bluetooth PC-Card is not present in your PC before upgrading

Your device

Type: Tektronix Bluetooth Protocol Analyzer

Firmware version: 9

This upgrade

Firmware version: 17

Baseband version: 930

Compilation date: 2001/06/14

Compilation time: 10:47:12

Figure 1-3: USB Update Flash screen

6. Browse again to where you unzipped the files, and run the setup.exe program.

7. The setup program asks if you want to remove the previous version of software. Follow the on-screen steps to remove the previous version. When prompted, restart your computer.

8. The installation procedure should automatically restart after your computer reboots. If not, run the setup.exe program to continue the installation process. Follow the on-screen prompts and restart you computer when prompted. You are now ready to operate your Bluetooth Protocol Analyzer.

Uninstalling DemoCard Software

If a Bluetooth DemoCard is installed on your computer, you must uninstall it before you can install the Bluetooth Protocol Analyzer.

To uninstall the Bluetooth DemoCard:

1. Insert the DemoCard.
2. Go to Settings/Control Panel in Windows.
3. Open the Add/Remove Programs icon.
4. Select Bluetooth DemoCard from the list, and follow the on-screen instructions.
5. Remove the DemoCard.
6. Restart your computer.

Uninstalling Earlier Versions of Bluetooth Software

To uninstall the earlier version of the Bluetooth software:

1. Go to Settings/Control Panel.
2. Open the Add/Remove Programs icon.
3. Select the Bluetooth software from the list, and follow the on-screen instructions. See the following note.

***NOTE**. As an alternate way to uninstall the older software, select the Uninstall Bluetooth Software Suite in the program folder.*

4. Restart your computer.

You can now install the Bluetooth Protocol Analyzer as described in *Installation Procedure for New Installs* on page 1-9.

Operating Basics

Operating Basics

This chapter describes the features and basic menus for the Bluetooth Data Collector and the Bluetooth Packet Analyzer applications.

Data Collector Operation

The purpose of the Bluetooth Data Collector is to monitor the Bluetooth piconet to which it is connected and to create a log containing all the baseband packets transmitted between the Bluetooth devices participating in the piconet. With the Data Collector, you can:

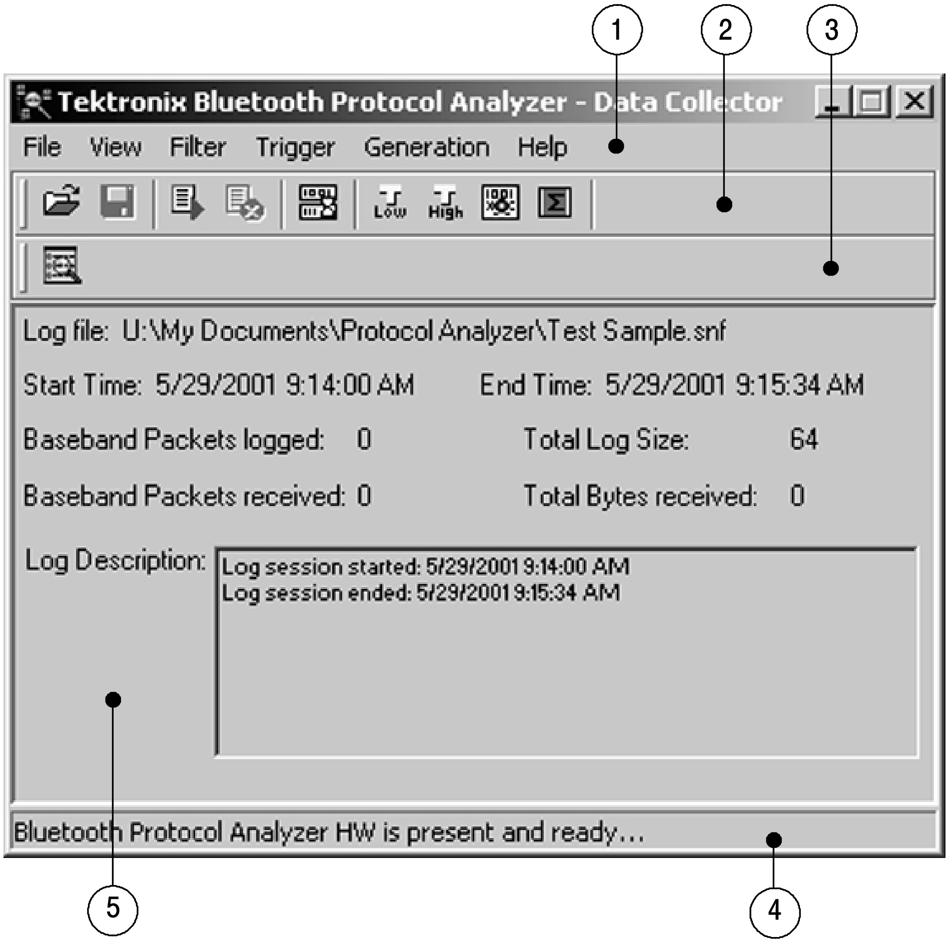
- Operate as a member of a piconet, as a stand-alone (independent) unit, or independent with data decryption
- Select the master or slave to which the Bluetooth Protocol Analyzer is synchronized
- Set the time for which the Protocol Analyzer tries to synchronize to a piconet master
- Capture all baseband packets transmitted within a Bluetooth piconet–including packets that are normally not visible for the host, such as retransmitted packets–and view the status of each packet and estimated clock and hop frequency
- Select any specified hopping pattern: Europe/USA, Japan, France, or Spain
- Transmit and receive on a single user-defined frequency
- Set a correlation value
- Turn data whitening on and off
- Output data to a log file or view as a real-time display
- Start or stop log sessions manually
- Enable data decryption in Piconet or Independent Mode.

- Display paging sequence in Independent Mode.
- Filter packets during data acquisition (prior to logging), such as ID, NULL, POLL, and Access Error packets
- Use high and low level trigger functions to log only the data in which you are interested.
- Generate known errors for testing and debugging your design.

NOTE*. When you use the Bluetooth Data Collector with Bluetooth Neighborhood, you must use the piconet mode (working as a participant in a piconet). When you use the Bluetooth Data Collector in the independent mode (working as a passive listener), you cannot use it with Bluetooth Neighborhood.*

Main Window

Figure 2-1 shows the main window of the Bluetooth Data Collector. Each of the major areas of the main window is described in the text associated with the number of the area.


1
2
3
Tektronix Bluetooth Protocol Analyzer - Data Collector
File View Filter Trigger Generation Help
Low
High
5
4

Log file: U:\My Documents\Protocol Analyzer\Test Sample.snf

Start Time: 5/29/2001 9:14:00 AM End Time: 5/29/2001 9:15:34 AM

Baseband Packets logged: 0 Total Log Size: 64

Baseband Packets received: 0 Total Bytes received: 0

Log Description: Log session started: 5/29/2001 9:14:00 AM
Log session ended: 5/29/2001 9:15:34 AM

Bluetooth Protocol Analyzer HW is present and ready...

Figure 2-1: Main window of the Data Collector

1. Menu bar. The menu bar contains the File, View, Filter, Trigger, Generation, and Help menus and their associated menu items.

   From the File menu, you can:

   - Open files (see *Open Old Log Session from Disk* on page 2-5)
   - Save Files (see *Save Current Log Session to Disk* on page 2-5)
   - Start a log session (see *Start New Log Session* on page 2-5)

- Stop a log session (see *Stop Current Log Session* on page 2-13)
- Quit the application

From the View menu, you can:

- Toggle Always on Top so that the Bluetooth Protocol Analyzer-Data Collector window appears on top of any other application windows
- Set default settings for the Data Collector by selecting Default Settings in the View menu

From the Filter menu, you can:

- Set up the data acquisition filter to remove unwanted baseband packets before the data is logged (see *Data Acquisition Filter* on page 2-13)
- Set up decryption. See the Decryption of Data diagram on page 2-33

From the Trigger menu, you can:

- Set the pretrigger and posttrigger buffer sizes (see *Pre- Post Trigger Setup* on page 2-14)
- Set high level trigger sequences for RFCOMM and SDP protocols (see *High Level Trigger* on page 2-25)
- Set low level trigger sequences for all protocols (see *Low Level Trigger* on page 2-15)

From the Generation menu, you can set error packet generation sequences for testing and debugging (see *Error Packet Generation* on page 2-27).

From the Help menu, you can:

- Connect to the Tektronix web site
- Connect to the Digianswer web site
- See information about the Bluetooth Data Collector, such as the version number and hardware BD address

2. Toolbar buttons. These buttons are shortcuts to many of the functions of the Bluetooth Data Collector. These buttons are described in *Data Collector Toolbar Buttons* on page 2-5.
3. Bluetooth Packet Analyzer button. When you have logged a new file or opened an old file from the Data Collector, clicking this button will open the corresponding file in the Bluetooth Packet Analyzer.
4. Status bar. Displays the status of the Bluetooth Data Collector.
5. Data window. This window displays information about the current log file: location, start and end times, number of baseband packets logged, log size, and date.

Data Collector Toolbar Buttons

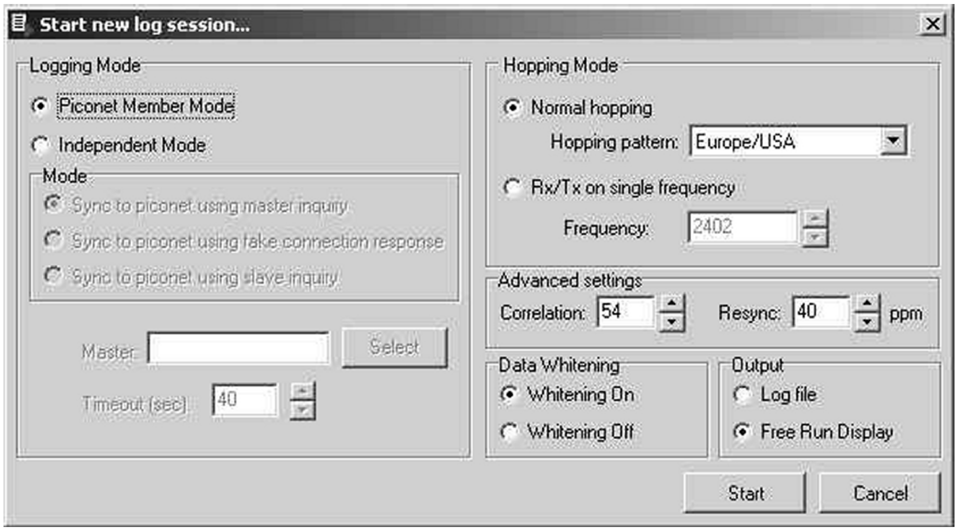
Open Old Log Session from Disk. Click this button to browse in Windows Explorer and open a previously stored log session.

Save Current Log Session to Disk. Click this button to save the current log session. The Bluetooth Protocol Analyzer will save three files of the log file with the following extensions:

- <Filename>.data contains only data. You can open files with this extension with the Bluetooth Packet Analyzer.
- <Filename>.desc contains text from the Log Description field in the Bluetooth Data Collector.
- <Filename>.snf are system files that the Bluetooth Data Collector uses to reference the log session, including the associated .data and .desc files.

Start New Log Session. Click this button to open the Start new log session... dialog box as shown in Figure 2-2.

The main sections of this dialog box are Logging Mode, Hopping Mode, Correlation, Data Whitening, and Output. Each of these sections are discussed in more detail.



Start new log session...

Logging Mode

● Piconet Member Mode

○ Independent Mode

Mode

● Sync to piconet using master inquiry

○ Sync to piconet using fake connection response

○ Sync to piconet using slave inquiry

Master:

Timeout (sec): 40

Hopping Mode

● Normal hopping

Hopping pattern: Europe/USA

○ Rx/Tx on single frequency

Frequency: 2402

Advanced settings

Correlation: 54

Resync: 40 ppm

Data Whitening

● Whitening On

○ Whitening Off

Output

○ Log file

● Free Run Display

Figure 2-2: Start new log session... dialog box

Logging Mode. Before you can start a new session, decide if you are going to operate the Bluetooth Protocol Analyzer as an active member of a piconet (either as a master or as a slave) or as a stand-alone unit that nonintrusively monitors data flowing across the piconet. The choices for logging mode are:

- Piconet Member Mode. Use this mode with the Bluetooth Neighborhood or HCI Terminal to set up the Bluetooth Protocol Analyzer as an active participant in the piconet. When you start a log session, the Data Collector logs all baseband packets sent from and received by your computer, whether the Bluetooth Protocol Analyzer is acting as a slave or a master.

- Independent Mode. Use this mode to set up the Bluetooth Protocol Analyzer as a stand-alone unit. The window shown in Figure 2-4 displays when synchronized in Independent Mode. You can select one of three kinds of synchronization modes:

  - Sync to piconet using master inquiry. In this mode the synchronization is obtained by performing an inquiry and using the clock information returned by the master to set the clock of the protocol analyzer. (You choose the master in the Select Master... dialog box that opens when you click the Select button. See Figure 2-3.)

In some Bluetooth devices, the clock drifts away when the device is not in connect mode; this synchronization mode can be troublesome if you want to monitor negotiations during the connect phase. The problem occurs because there are often several seconds of delay from the time when the protocol analyzer obtains the master clock information until the master actually connects to the slave. Likewise, if the inquiry scan mode on the Bluetooth device is not implemented or disabled during the connection, this mode cannot be used for synchronization. See *Resync* on page 2-12.

- Sync to piconet using a fake connection response. This mode can only be used during the connect phase, when the piconet master connects to a new slave. The protocol analyzer operates as if it were the slave unit chosen in the Select Slave... dialog box (see Figure 2-3) and obtains the master clock information by initiating a new connection as if it were that slave. Immediately after the clock information is retrieved, the protocol analyzer stops transmitting, and the piconet master continues the connection attempt with the true slave.

NOTE*. The HCI Terminal application provides user control of the BPA100 in piconet member mode. See the* HCI Terminal *topic on page 2-35.*

- Sync to piconet using slave inquiry. This mode can only be used during the connect phase and is based on the same principle as the method mentioned above in *Sync to piconet using fake connection response*. Instead of pretending to be the slave unit chosen in the Select Slave... dialog box (see Figure 2-3), the protocol analyzer listens for the clock information sent in the connect phase to the new piconet slave, and, therefore, does not interfere with the piconet in any way. To catch the clock information on the right frequency, it is necessary to obtain the slave clock. This is done by performing an inquiry to the slave.

Click Select in the Start new log session... dialog box (see Figure 2-2) to select a master or slave. The Select Master... or Select Slave... dialog box opens. Refer to Figure 2-3.

In the Inquiry Timeout dialog box, you can select how long the Bluetooth Protocol Analyzer performs the inquiry process. The default time is 12 seconds. However, you can set the time from 2 seconds to 60 seconds.

In the Inquiry Access Code dialog box, you can set an inquiry access code (IAC). There are 64 IACs. The default is the General IAC (GIAC) which is 0x9E8B33. The remaining 63 access codes are Dedicated IACs (DIACs). You can set any of the 64 IACs. Although the GIAC is normally used, you can use a DIAC in certain instances.

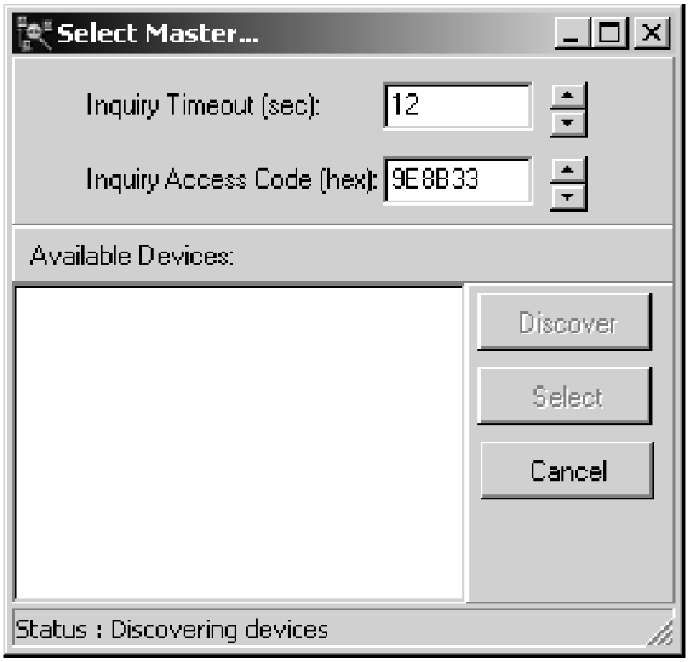
For example, a group of users might agree to set their devices to a specific DIAC to make their devices easier to discover in an environment with many Bluetooth devices.

Click the Discover button to carry out device discovery and display a list of all active Bluetooth devices within range.

Click the Select button to synchronize to the device that you have highlighted. Close the Select Master... dialog box after selecting the device to which you want to synchronize.

In the Start new log session... dialog box (see Figure 2-2) you use the Timeout (sec) field to set the number of seconds allowed to pass after synchronization to the piconet when there is no activity in the piconet. On time-out, the Bluetooth Protocol Analyzer will lose synchronization and display the message, *Out of sync with piconet!*.

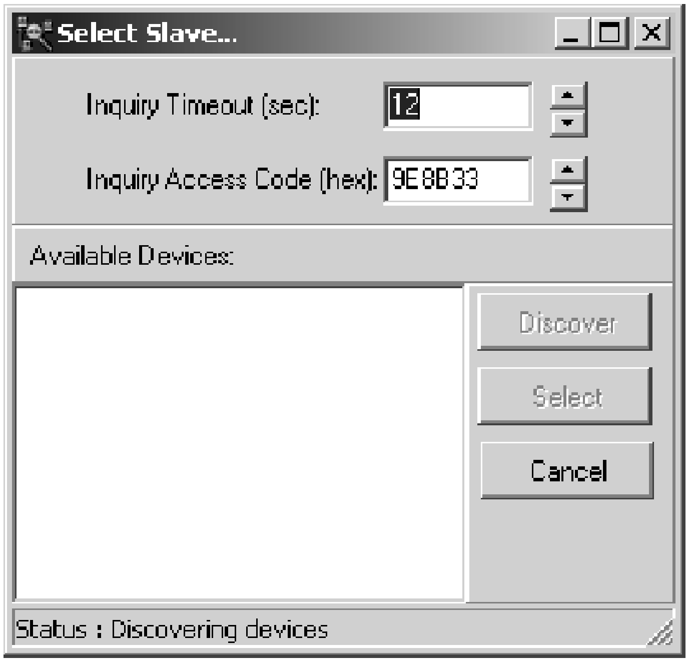
***NOTE**. When you use the Bluetooth Data Collector with Bluetooth Neighborhood, you must use the piconet mode (working as a participant in a piconet). When you use the Bluetooth Data Collector in the independent mode working as a passive listener, you cannot use it with Bluetooth Neighborhood.*



Inquiry Timeout (sec): 12

Inquiry Access Code (hex): 9E8B33

Status : Discovering devices



Inquiry Timeout (sec): 12

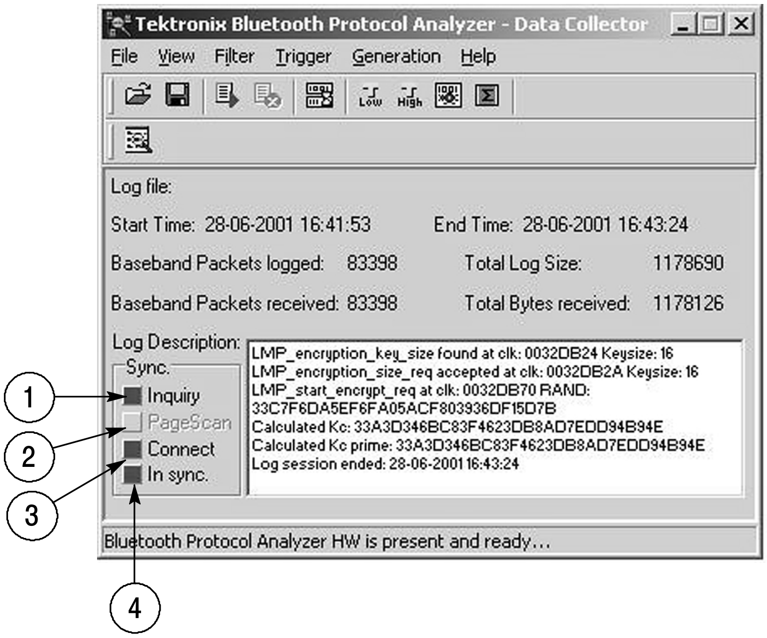
Inquiry Access Code (hex): 9E8B33

Status : Discovering devices

Figure 2-3: Select Master... and Select Slave... dialog boxes

Sync Indication Panel. When you select Independent Mode as the logging mode, the Data Collector screen displays a Sync Indication panel similar to Figure 2–4. The four indicators have the following functionality:

- Indicator 1 = Inquiry. It is colored Green when the BPA100 starts the inquiry procedure during master inquiry or slave inquiry. The indicator is colored Red if the unit that is inquired does not answer within a 60-second timeout.
- Indicator 2 = PageScan. Indicator is colored Green when the BPA100 enters the Page scan part of the synchronization procedure. It is therefore only present if slave inquiry or fake connection is chosen. No timeout is present in the part of the synchronization procedure, therefore the user can only stop the synchronization by clicking the stop current log session button on the toolbar.
- Indicator 3 = Connect. Indicator is colored Green when the BPA100 enters the channel hopping sequence (BPA100 searches for first traffic on the piconet). If no traffic is recorded, the indicator is colored Red, and the synchronization failed.
- Indicator 4 = In sync. Indicator is colored Green when the first packet is received on the channel hopping sequence. If the synchronization to the piconet is lost (41-second timeout) this indicator is colored Red which means that the synchronization to the piconet is lost. When this happens, a screen similar to Figure 2–5 displays.



Log file:

Start Time: 28-06-2001 16:41:53 End Time: 28-06-2001 16:43:24

Baseband Packets logged: 83398 Total Log Size: 1178690

Baseband Packets received: 83398 Total Bytes received: 1178126

Log Description:

Sync.

■ Inquiry

□ PageScan

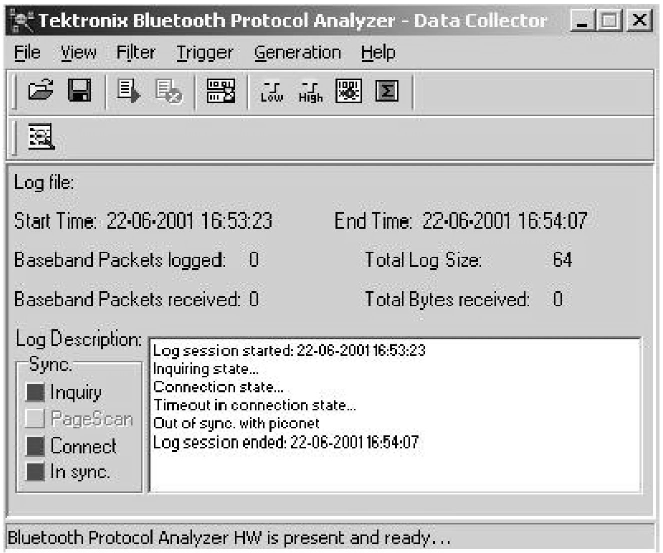
■ Connect

■ In sync.

```
LMP_encryption_key_size found at clk: 0032DB24 Keysize: 16
LMP_encryption_size_req accepted at clk: 0032DB2A Keysize: 16
LMP_start_encrypt_req at clk: 0032DB70 RAND:
33C7F6DA5EF6FA05ACF803936DF15D7B
Calculated Kc: 33A3D346BC83F4623DB8AD7EDD94B94E
Calculated Kc prime: 33A3D346BC83F4623DB8AD7EDD94B94E
Log session ended: 28-06-2001 16:43:24
```

Bluetooth Protocol Analyzer HW is present and ready…

Figure 2-4: Sync view in Independent Mode with values



Log file:

Start Time: 22-06-2001 16:53:23 End Time: 22-06-2001 16:54:07

Baseband Packets logged: 0 Total Log Size: 64

Baseband Packets received: 0 Total Bytes received: 0

Log Description:

Sync.

■ Inquiry

□ PageScan

■ Connect

■ In sync.

```
Log session started: 22-06-2001 16:53:23
Inquiring state...
Connection state...
Timeout in connection state...
Out of sync. with piconet
Log session ended: 22-06-2001 16:54:07
```

Bluetooth Protocol Analyzer HW is present and ready…

Figure 2-5: Out of Sync view in Independent Mode

Hopping Mode. In this section of the Start new log session... dialog box, you can select either Normal hopping or Rx/Tx on single frequency. If you choose Normal hopping, you must also select the hopping pattern for the geographical area you want (Europe/USA, France, Spain or Japan).

Or you can select Rx/Tx on single-frequency and specify the desired frequency (from 2402 MHz to 2480 MHz). This mode is useful for testing and debugging.

***NOTE**. To meet FCC regulations, the transmit power is reduced from 20 dBm to 0 dBm when operating in the single-frequency mode.*

Correlation. You can set a correlation value in this section of the Start new log session... dialog box. The correlation value sets the number of bits in the sync word of each received packet that must be matched for the packet to be valid. Normally, the radio uses 54 to 64 bits correlation. The default value is 54. The value can range from 40 to 64.

Resync. You can set a resync value in this section of the Start new log session... dialog box. See Figure 2-2. The resync value sets the drift in parts per million. If synchronization is lost during a connection, for example when the link enters Park, Sniff or Hold mode, user can enter the drift in PPM. Instead of the normal limit of 250 PPM that a device may drift in Park, Sniff or Hold mode, the user can force the BPA100 not to use ”window search” by setting the resync drift to 40 PPM (default). This is useful if the user knows that the device has a small drift. This ensures that no packets are lost because of the window search.

Data Whitening. Data whitening can be turned on or off. By default, the function is set to on, which is normal operation for Bluetooth devices. Data whitening encrypts all data packets that are sent between Bluetooth devices on a piconet to remove DC bias in the transmitted data. However, for test purposes, you can turn off data whitening. In this test situation all devices must have whitening turned off, or you will get scrambled data.

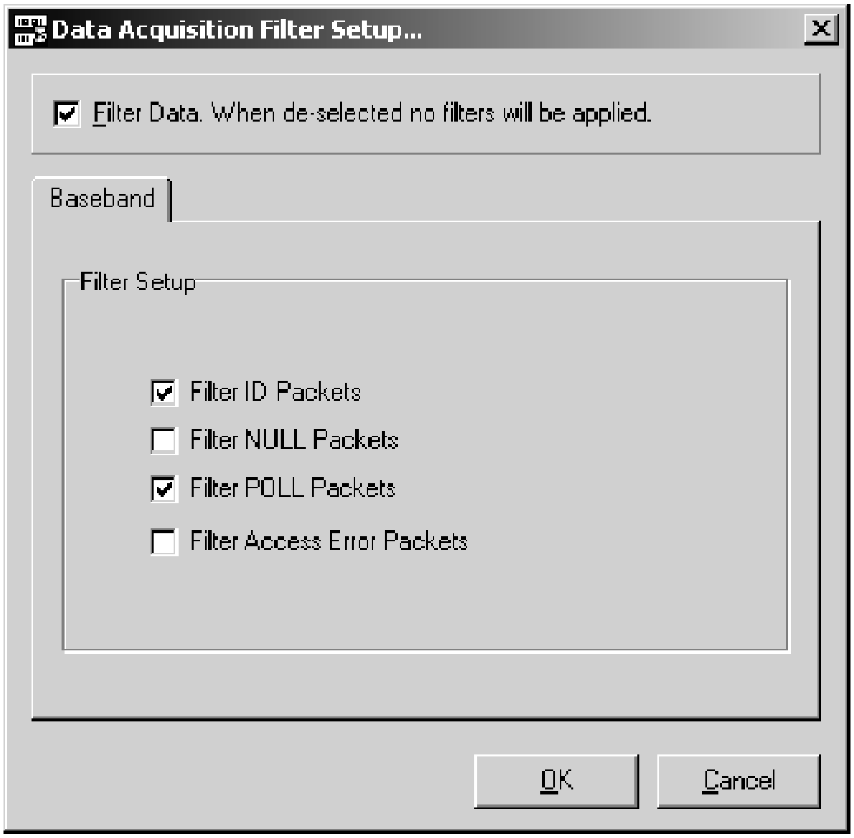
Output. In this section of the Start new log session... dialog box, you have two choices for where to send the data of the log session. You can send the output of your log session to a log file, which you can open later with the Packet Analyzer. Or if you select Free Run Display (see Figure 2-2 on page 2-6), you can send the data directly to the list view field in the Bluetooth Packet Analyzer main window (see Figure 2-18 on page 2-37). When Free Run Display is selected, the data is also sent to a log file.

Free Run Display. Allows you to continuously monitor the latest session transactions with real-time screen updates while logging directly to the hard disk of the PC. This includes the display of both encrypted and decrypted data.

***NOTE**. Before starting a new log session using free run display (see Figure 2-2 on page 2-6), you must first close the Bluetooth Packet Analyzer application, if it is open.*

Stop Current Log Session. Click this button to stop the current log session. The Data Collector main window will now display information on the start and end times of the log session, number of baseband packets logged, and log size.

Data Acquisition Filter. Click this button to display the Data Acquisition Filter Setup... dialog box. See Figure 2-6. You can set up this filter to remove the following baseband packets before the data is logged: ID packets, NULL packets, POLL packets, and Access Error packets.


Data Acquisition Filter Setup...

☑ Filter Data. When de-selected no filters will be applied.

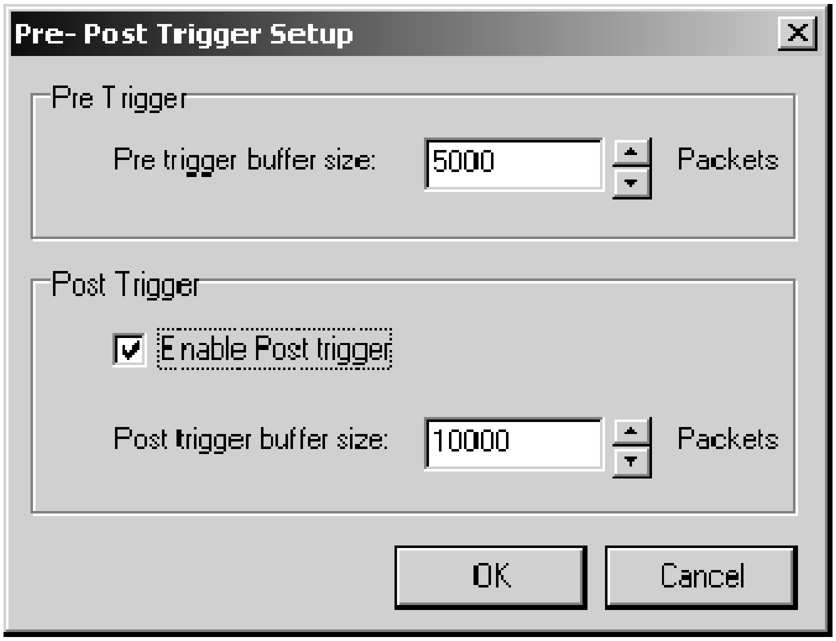
☑ Filter ID Packets
☐ Filter NULL Packets
☑ Filter POLL Packets
☐ Filter Access Error Packets

Figure 2-6: Data Acquisition Filter Setup... dialog box

Pre- Post Trigger Setup.

NOTE. *This is a menu item under the Trigger menu. There is no corresponding toolbar button for this function.*

Select Pre- Post Trigger from the Trigger menu to display the dialog box for setting pretrigger and posttrigger buffer sizes. See Figure 2–7. You use this dialog box to set how many packets are saved prior to the trigger event (0 to 100,000) and how many packets are saved after the trigger event (user-defined value). If you do not check the Enable Post trigger box, posttrigger data is saved until you manually stop the logging or the hard disk becomes full.



Pre- Post Trigger Setup

Pre Trigger

Pre trigger buffer size: 5000 Packets

Post Trigger

☑ Enable Post trigger

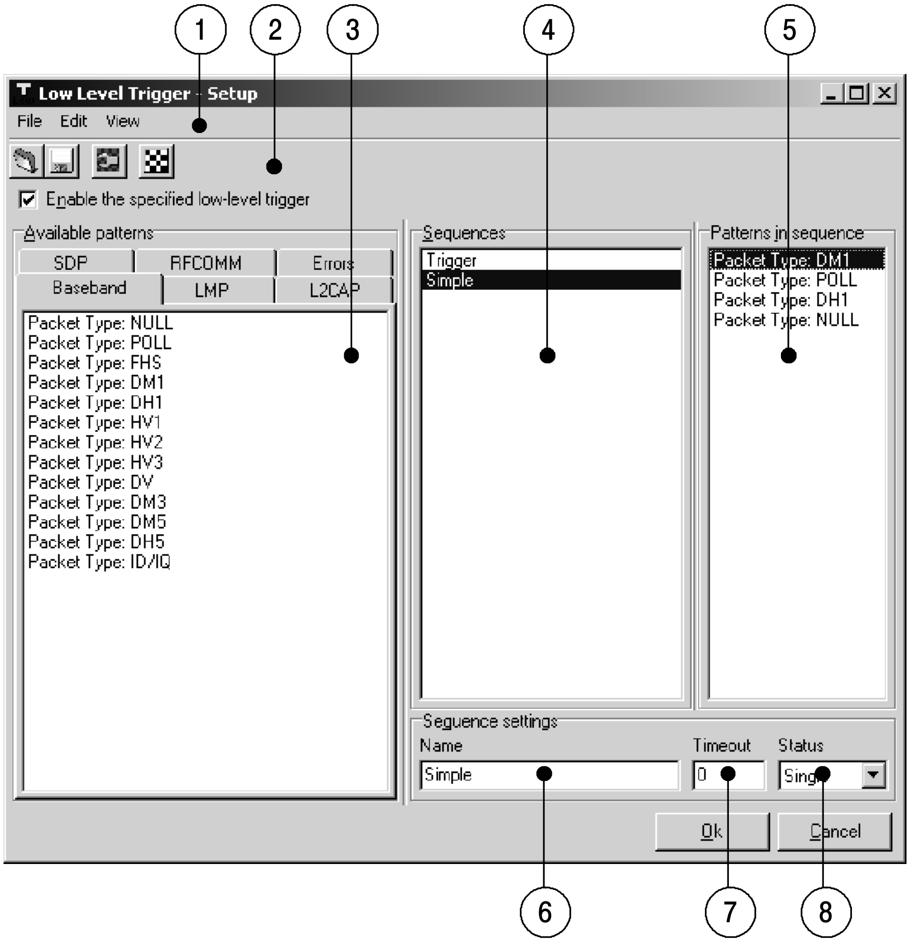
Post trigger buffer size: 10000 Packets

Figure 2-7: Pre- Post Trigger Setup dialog box

Low Level Trigger

Click this button to display the Low Level Trigger-Setup window. See Figure 2-8. You use this window to set up the low level triggers. Each of the major areas of the Low Level Trigger-Setup window is described in the text associated with the number of the area.

***NOTE**. Due to hardware limitations, you are allowed only 10 hardware patterns (slots 0 through 9) for low level triggers and/or error packet generation. Also see* Patterns in hardware property sheet *on page 2-24. This means you cannot use both low level trigger and error packet generation functions simultaneously.*



☑ Enable the specified low-level trigger

Available patterns

Packet Type: NULL
Packet Type: POLL
Packet Type: FHS
Packet Type: DM1
Packet Type: DH1
Packet Type: HV1
Packet Type: HV2
Packet Type: HV3
Packet Type: DV
Packet Type: DM3
Packet Type: DM5
Packet Type: DH5
Packet Type: ID/IQ

Sequences

Trigger
Simple

Patterns in sequence

Packet Type: DM1
Packet Type: POLL
Packet Type: DH1
Packet Type: NULL

Sequence settings

Name: Simple
Timeout: 0
Status: Sing

Figure 2-8: Low Level Trigger-Setup window

1. Menu bar. The menu bar contains the File, Edit, and View menus and their associated menu items. From the File menu, you can open an existing Data Collector trigger setup file (*.dct) or save the current trigger setup as an *.dct file.

   From the Edit menu, you can display the Customize Pattern dialog box.

   From the View menu, you can display the Patterns in hardware property sheet.

2. Toolbar buttons. These buttons are shortcuts to menu items in the Menu bar. Each of the buttons has a corresponding menu item in the menu bar. These buttons are described in *Low Level Trigger Toolbar Buttons* on page 2-19.

3. Available patterns. This field displays the available patterns for the selected tab. You can add a pattern to a sequence in one of three ways:

   - Double-click the pattern you want to add to the sequence
   - Highlight the pattern you want to add to the sequence, and then right-click to display a context-sensitive menu
   - Drag the pattern you want to add to the sequence to the Patterns in sequence field

   There are ten hardware slots into which you can load patterns. See *Patterns in hardware property sheet* on page 2-24.

4. Sequences. This field displays the sequences that you have created. You can create a maximum of four sequences, each of which can contain a maximum of four patterns. The default sequence is named Trigger. As you create additional sequences, they will automatically be named Trigger1, Trigger2, and Trigger3.

   Each sequence is a potential trigger. Whichever sequence is found first triggers the Bluetooth Data Collector to begin logging. Occurrences of the remaining sequences are indicated in color and function as markers in the Bluetooth Packet Analyzer display.

   The color codes are as follows:

   - Yellow indicates a pattern in an active sequence
   - Green indicates the final pattern (low and high-level trigger packets)
   - Red indicates a time-out

For example, the following two sequences are set up:

Sequence1 (Status set to Single)
LMP_detach
NULL

Sequence2 (Status set to Single)
LMP_host_connection_request
LMP_accepted

If you monitor a connection establishment followed by a connection detachment, sequence2 will be found first and will be the trigger. Sequence1 will function as a marker.

5. Patterns in sequence. This field shows the patterns that are contained in the sequence that is highlighted in the Sequence field. You can add four patterns to a sequence (see *Available Patterns* on page 2-17).
6. Name. This field displays the name of the sequence that is highlighted in the Sequences field. You can use this field to change the default name of a sequence that you have created. Additionally, the settings of the Timeout, Status, and Count fields are applied to the sequence whose name is displayed in this field.
7. Timeout. You use this field to control how long the application looks for the next pattern in a sequence. Enter the value as the number of Bluetooth time units. A Bluetooth time unit is 625 μs. The range for this field is 0 to 65535 time units. If you enter 0, you disable the time-out. If a time-out precludes a sequence from completing, a red marker is indicated in the Bluetooth Packet Analyzer list view and the sequence is reset.
8. Status. You use this field to control the status of each of the sequences that you have created. This is a different field from Status in the Customize pattern dialog box. The following four status selections are available:

   - Off. When selected, the highlighted sequence is disabled and will not be recognized by the Bluetooth Protocol Analyzer.
   - Single. When single is selected, only the first occurring sequence whose patterns occur in their listed order will be marked in the Bluetooth Packet Analyzer display.

- Repeat. Whenever the patterns in the specified sequence occur in order, they will be marked in the Bluetooth Protocol Analyzer display.
- Number. When you select number as the status, an additional field called Count is displayed. The value in this field determines the number of times the sequence is marked. You can enter a value from 2 through 200. In all cases, the first sequence to complete triggers the Bluetooth Data Collector, and the following sequences are marked in the Bluetooth Packet Analyzer display.

Low Level Trigger Toolbar Buttons

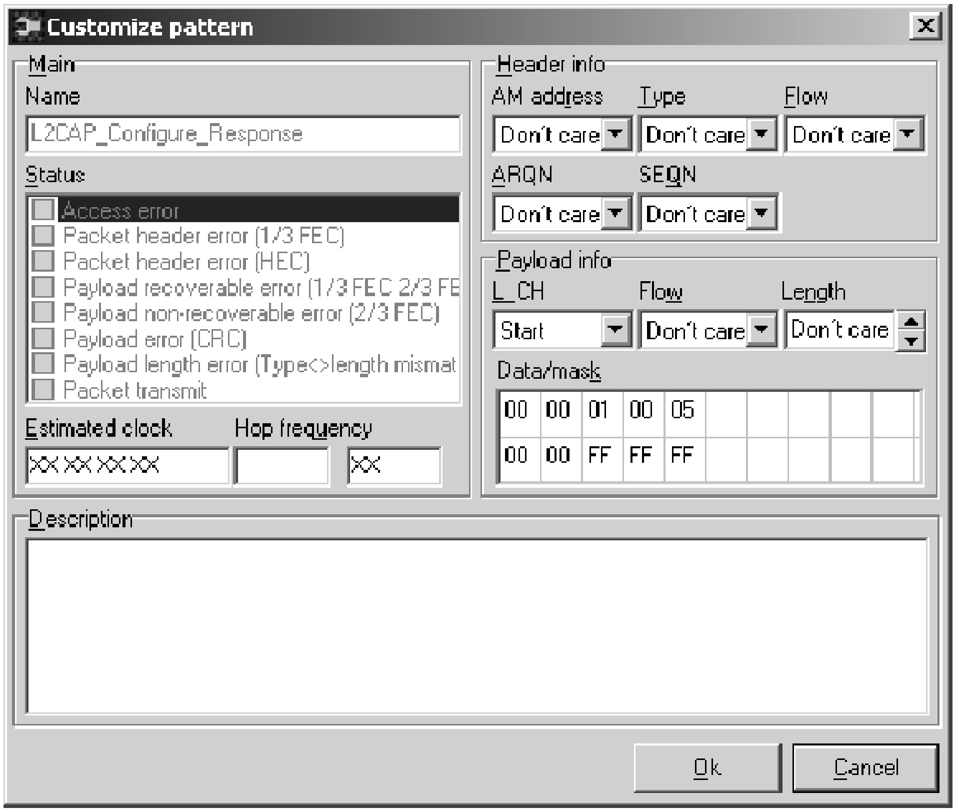
Load Workspace. Click this button to display the Open dialog box that allows you to browse and open a Data Collector trigger setup file (*.dct).

Save Workspace. Click this button to display the Save As dialog box that allows you to browse and save a Data Collector trigger setup file (*.dct).

Customize Pattern. Click this button to access the Customize Pattern dialog box in which you can set up advanced triggering parameters. See Figure 2-9.

To activate the Customize Pattern button, you must do the following in the Low Level Trigger-Setup dialog box (see Figure 2-8 on page 2-16):

- Check the Enable the specified low-level trigger box.
- Set up one or more sequences containing one or more patterns.
- Select the sequence containing the pattern that you want to modify.
- Select the pattern that you want to modify.



Name: L2CAP_Configure_Response

Status

- □ Access error
- □ Packet header error (1/3 FEC)
- □ Packet header error (HEC)
- □ Payload recoverable error (1/3 FEC 2/3 FE
- □ Payload non-recoverable error (2/3 FEC)
- □ Payload error (CRC)
- □ Payload length error (Type<>length mismat
- □ Packet transmit

Estimated clock: XX XX XX XX

Hop frequency: XX

AM address: Don't care
Type: Don't care
Flow: Don't care
ARQN: Don't care
SEQN: Don't care

L_CH: Start
Flow: Don't care
Length: Don't care

Data/mask

00	00	01	00	05					
00	00	FF	FF	FF					

Figure 2-9: Customize Pattern dialog box

Other methods of accessing the Customize pattern dialog box are as follows:

- Double-click a pattern in the Patterns in sequence field in the Low Level Trigger-Setup dialog box (see Figure 2-8 on page 2-16).
- Highlight a pattern in the Patterns in sequence field in the Low Level Trigger-Setup dialog box; right-click in the sequence field to display a context-sensitive menu. Select Customize pattern from the menu.

The fields in the Customize Pattern dialog box are described in the following text:

Name. This field displays the name of the pattern that you selected to customize in the Low Level Trigger-Setup dialog box (see Figure 2-8 on page 2-16).

Status. This field contains information about the status of the packet. This is a different field from Status in the Low Level Trigger-Setup dialog box. Here Status indicates whether the packet is an RX or TX packet. For a receive packet, this field also may contain information about errors that were in the packet (for example, Header Errors and Payload Errors). There are no restrictions in what can be specified, so it is possible to specify a trigger on a TX packet with access error, although this is not a combination that can occur. You can also specify the bits to be "don't care".

All the fields in Customize pattern dialog box are used to set conditions for trigger to occur. In the Status field you can set some conditions like trigger only if an error occurs. The following options are available in the Status field:

- Access error
- Packet header error (1/3 FEC)
- Packet header error (HEC)
- Payload recoverable error
- Payload non-recoverable error
- Payload error
- Payload length error
- Packet transmit

By right-clicking you can enable and set the condition or make the condition "don't care." For example, if you select the the third option, then trigger on that pattern occurs only if there is an HEC error in that pattern. If you select the eighth option, trigger occurs only if that pattern is transmitted.

Estimated Clock. This is the Bluetooth clock for the master used in the piconet. X specifies that four bits are “don’t care”. For example, XXXXXXXX causes the entire estimated clock is to be ignored in the triggering.

Hop Frequency. In this two-part field, you can enter a specific frequency. In addition to the frequency, the channel is displayed (on the right). The mapping from frequency to channel is (Freq = 2402+Channel), and the mapping goes both ways. For example, if you specified channel 10, the frequency field automatically displays 2412. You can also select “don’t care” for these bits.

AM Address. This field sets the Active Member (AM) address. This address is used to access different members in the piconet. Three bits are used for this address, that is, eight different AM addresses are available. AM_ADDR = 0 is used for broadcast. You can also select “don’t care” for these bits.

Type. This field specifies the packet type. Four bits are used for the packet type, that is, 16 different Packet types are available. You can specify only the packets that are not reserved. You can also select “don’t care” for these bits.

Flow. One bit is used for flow control in the header. Flow = 0 means STOP; Flow = 1 means GO. You can also select “don’t care” for this bit.

ARQN. One bit is used for acknowledgement of the last transmission. If a packet is received correctly, the ARQN bit is set to 1 in the return packet. You can also select “don’t care” for this bit.

SEQN. The SEQN is a sequential numbering used to detect retransmission. You can also select “don’t care” for this bit.

L_CH. This field specifies the Logical Channel. This field is two bits and is used to indicate if the packet is a LMP message or a L2CAP fragment.

Flow. This flow bit is used to control flow on the L2CAP level. One bit is used for flow control in the payload. Flow = 0 means STOP; Flow = 1 means GO. You can also select “don’t care” for this bit.

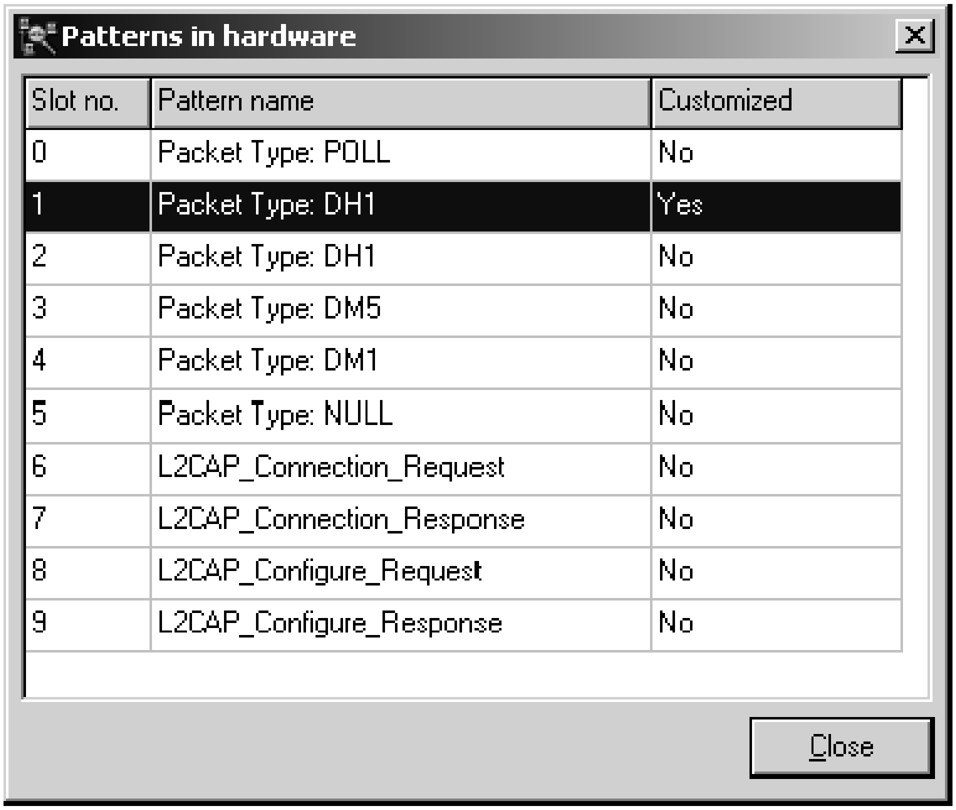
Length. This field allows you to select a specific length to trigger on. The length can be from 0–339, and you can also select “don’t care”.

Data/Mask. This field specifies the payload data (the first row) and the mask that is used with the data (the second row). A mask of FF will mask in the whole byte and a mask of 00 will mask out the whole byte. The position of the mask and Data is linked together so that the value in data index 1 links to the mask at mask index 1 and so on.

Description. You can use this field to enter additional information (notes) about the specified pattern.

Patterns in Hardware. Click this button to display the Patterns in hardware property sheet, which shows information about the patterns you have loaded into hardware. See Figure 2–10.

***NOTE**. Due to hardware limitations, you are only allowed 10 hardware patterns (slots 0 through 9) for low level triggers and/or error packet generation. This means you cannot use both low level trigger and error packet generation functions simultaneously.*

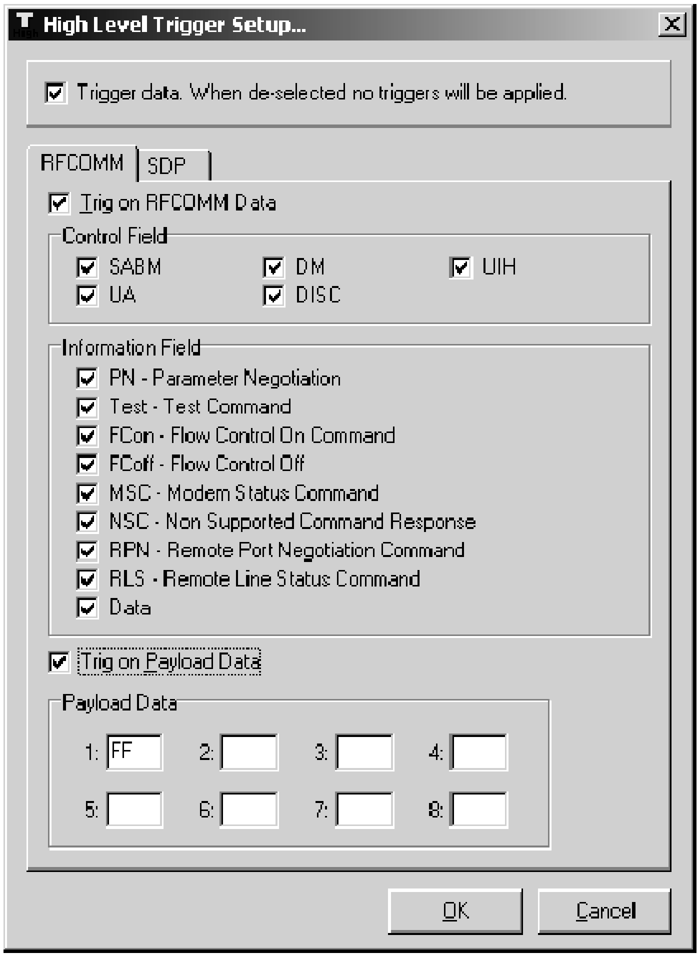


Slot no.	Pattern name	Customized
0	Packet Type: POLL	No
1	Packet Type: DH1	Yes
2	Packet Type: DH1	No
3	Packet Type: DM5	No
4	Packet Type: DM1	No
5	Packet Type: NULL	No
6	L2CAP_Connection_Request	No
7	L2CAP_Connection_Response	No
8	L2CAP_Configure_Request	No
9	L2CAP_Configure_Response	No

Figure 2-10: Patterns in hardware property sheet

High Level Trigger

Select High level Trigger... from the Trigger menus to open the High Level Trigger Setup... dialog box. See Figure 2-11. You use this dialog box to set up high level triggers for the RFCOMM protocol and the Service Discovery Protocol (SDP).



☑ Trigger data. When de-selected no triggers will be applied.

☑ Trig on RFCOMM Data

Control Field

☑ SABM ☑ DM ☑ UIH
☑ UA ☑ DISC

Information Field

- ☑ PN - Parameter Negotiation
- ☑ Test - Test Command
- ☑ FCon - Flow Control On Command
- ☑ FCoff - Flow Control Off
- ☑ MSC - Modem Status Command
- ☑ NSC - Non Supported Command Response
- ☑ RPN - Remote Port Negotiation Command
- ☑ RLS - Remote Line Status Command
- ☑ Data

☑ Trig on Payload Data

Payload Data

1:	FF	2:		3:		4:	
5:		6:		7:		8:	

Figure 2-11: High Level Trigger Setup... dialog box

To setup and/or trigger on RFCOMM or SDP protocols, you must check the Trigger data check box near the top of the dialog box.

When you click the RFCOMM tab and select the Trig on RFCOMM Data check box, you have the following information fields from which you can select: SABM, UA, DM. DSC, and UIH. If you check UIH, additional information fields become active.

You can also select Trig on Payload Data to set up a trigger on the first 8 bytes of payload data. (Values for each byte are 0 through FF.) Empty fields mean Don't Care. For RFCOMM, the Payload data starts from the second byte of the RFCOMM information field; for SDP, the Payload data starts from the first byte of the SDP parameter data part.

When you click the SDP tab in the dialog box and select the Trig on SDP Data box, you can set up triggers for SDP_PDU (Protocol Data Unit) transactions, such as Trig on 0x01 SDP_ErrorResponse between the server and the client. You can select the PDUs on which you want to trigger by selecting the box next to the SDP_PDUs in the list displayed in the PDUs section of the dialog box. You can also select Trig on Payload Data to set up a trigger on the first 8 bytes of payload data. (Values for each byte are 0 through FF.)

Differences between High Level and Low Level Triggers

The main difference between Low Level Trigger (LLT) and High Level Trigger (HLT) is the option to customize the pattern and the ability to trigger at all layers of Bluetooth stack. Some of the other features are:

- CIDs (Channel Identifiers) are logical endpoints used in the L2CAP layer to connect with other devices and are vendor-specific. From 0x0040–0xffff, a vendor can implement as needed.
- If you use a Bluetooth device other than Digianswer, the vendor might have used a different CID in the L2CAP layer.
- For Digianswer, the SDP layer uses 0x0040 and the RFCOMM layer uses 0x0041. This information is available in the Description part of the Customize pattern dialog in LLT. This information also is found in the Packet Analyzer when doing service discovery for SDP and business card exchange for RFCOMM.
- If a Bluetooth device has a different CID for SDP and RFCOMM, you need to find the CID values and change them in Customize pattern dialog in order to trigger on that pattern. For example, if the Ericsson™ SDP CID is 0x0FFF then you have to change the value in Customize Pattern Data field. You do not need to change the mask value.

- For Digianswer:

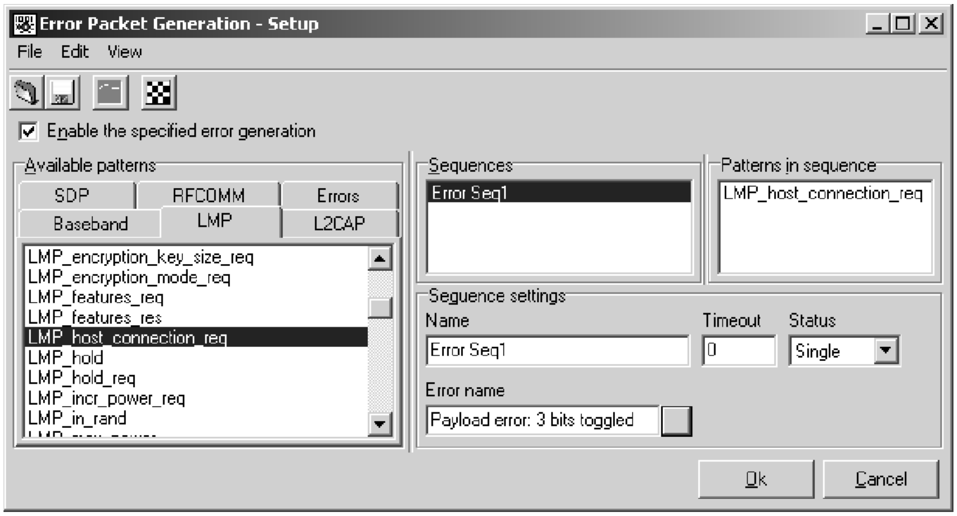
  DATA : 00 00 41 00 01 73
  MASK : 00 00 FF FF 01 FF

- For other vendor if CID is 0x0FFF

  DATA : 00 00 FF 0F 01 73
  MASK : 00 00 FF FF 01 FF

- In HLT the application can find the CID value of the other device. This occurs when both devices exchange the CID value before establishing a L2CAP connection between the two devices. It is important for the HLT to have a high pre trigger buffer value set so that the triggers are marked when the log file is loaded. This is the reason HLT sometimes fails to indicate or mark, although it actually triggers at the specified pattern.

Error Packet Generation. Click this button to display the Error Packet Generation-Setup window. See Figure 2–12. This window allows you to generate error packets for testing the handling of errors and possible retransmissions. You can use error generation to cross-check error-correcting algorithms, such as FEC, HEC, and CRC. You can generate error packets for any baseband packet, such as DM1, DM3, POLL, and so on. Errors are introduced by individual bits in the header, payload, or in a custom-defined bit position of the packet.



☑ Enable the specified error generation

LMP_encryption_key_size_req
LMP_encryption_mode_req
LMP_features_req
LMP_features_res
LMP_host_connection_req
LMP_hold
LMP_hold_req
LMP_incr_power_req
LMP_in_rand

Sequences: Error Seq1

Patterns in sequence: LMP_host_connection_req

Name: Error Seq1
Timeout: 0
Status: Single
Error name: Payload error: 3 bits toggled

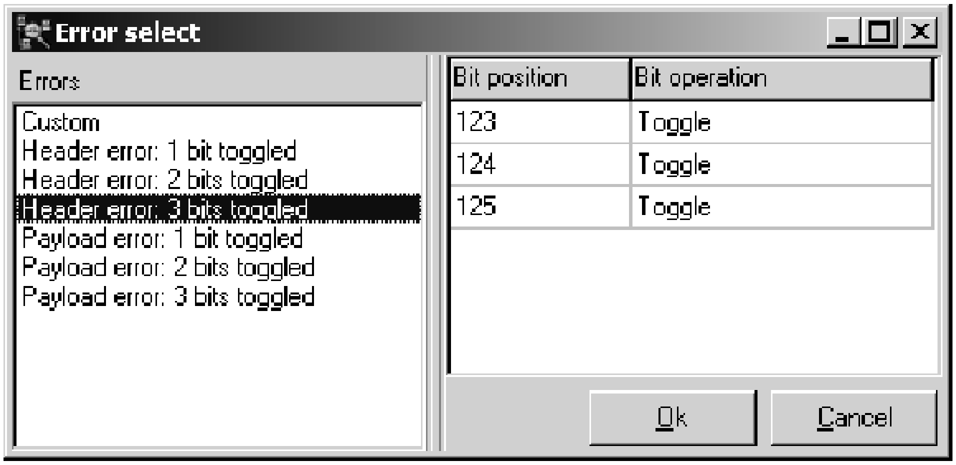
Figure 2-12: Error Packet Generation-Setup window

***NOTE**. Error packet generation and low level triggering settings (see page 2-15) are similar functions. However, due to hardware limits, you cannot use both functions at the same time. Also, see Patterns in hardware property sheet on page 2-24.*

With the exception of the the Error name field, which is explained below, this window is identical to the Low Level Trigger-Setup window (see Figure 2-8 on page 2-16). Refer to *Low Level Trigger* on page 2-15 for detailed information.

To generate an error in a sequence you have created, select the sequence in which you want to insert an error, and then click the button adjacent to the Error name field. This displays the Error select dialog box shown in Figure 2-13.

***NOTE**. If you have more than one pattern in the sequence for which you are generating an error, the error is sent with the last pattern in the sequence.*



Custom
Header error: 1 bit toggled
Header error: 2 bits toggled
Header error: 3 bits toggled
Payload error: 1 bit toggled
Payload error: 2 bits toggled
Payload error: 3 bits toggled

Bit position	Bit operation
123	Toggle
124	Toggle
125	Toggle

Figure 2-13: Error Select dialog box

You can select from one of the defined header or payload errors, or you can select Custom and enter a bit position and bit operation of your choosing.

NOTE. *Error generation on packets that contain payload data may not have errors introduced into the access code or into the first few bytes of the header. This is because the first few bytes of the packet will already have been transmitted by the time the error packet generator recognizes this packet as one in which to introduce errors.*

Header error. A header with a 1-bit error should be recoverable by devices receiving the error packet. A 2- or 3-bit error results in an unrecoverable error in the receiving device. Packets with recovered errors are indicated in green text in the list window of the Bluetooth Packet Analyzer; unrecovered errors are displayed in red text.

Payload error. CRC is used for error checking the payload. Similar to header errors, a 1-bit error is recoverable; 2- and 3-bit errors are not recoverable. Bit positions 126 and 127 correspond to the L_CH of the payload header format. See Figure 2–14.

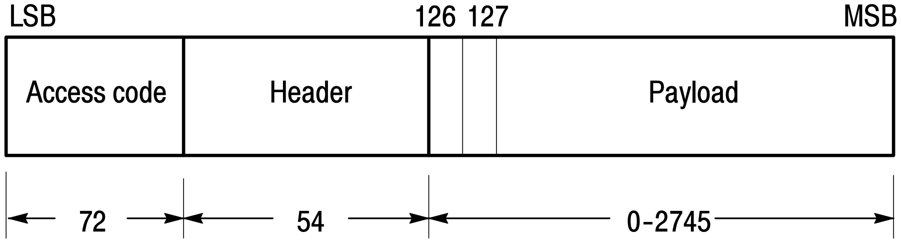


Figure 2-14: Standard packet format

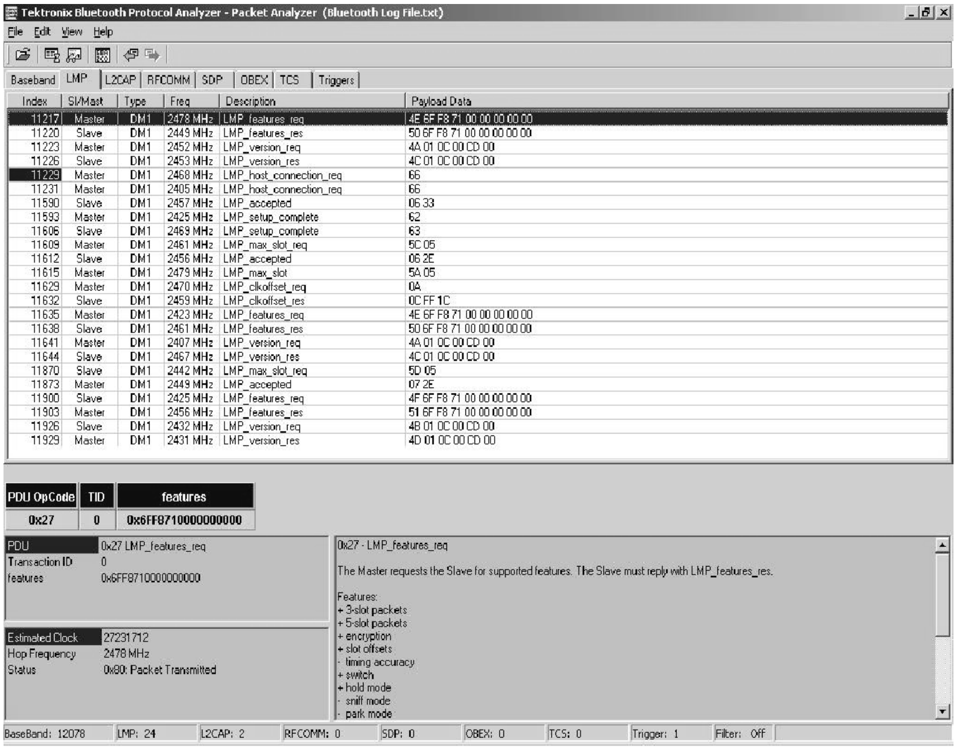
When generating a 2- or 3-bit error, it is recommended that you do not use the Repeat status (in the Error Packet Generation-Setup window), since this will result in a continuous, unrecoverable error. Instead, use the Number status and set the count to a desired value (for example, set the count to 5).

Custom error. To enter the bit operation for a custom error, click the Bit operation field to activate a pull down menu from which you can choose Forced 1, Forced 0, or Toggle as the bit operation. It is recommended that you use Toggle instead of Forced 1 or Forced 0.

Example of a Generated Error

In Figure 2-12 on page 2-28, the Error Packet Generation-Setup window was used to create a sequence named Error Seq1 that contained an LMP_host_connection_req pattern. A Payload error with 3 bits toggled was set to be transmitted with this pattern. The status was set to Single, which resulted in the error being transmitted one time. Figure 2-15 shows the Bluetooth Packet Analyzer display resulting from transmitting the error.

Under the index tab, 11229 is highlighted (in blue in the application). This indicates an error was transmitted. Following this error, 11231 shows that the LMP_host_connection_req pattern was transmitted again but without the error. (For detailed information about the Bluetooth Packet Analyzer, refer to *Packet Analyzer Operation* on page 2-36.)


Tektronix Bluetooth Protocol Analyzer - Packet Analyzer (Bluetooth Log File.txt)
File
Edit
View
Help
Baseband
LMP
L2CAP
RFCOMM
SDP
OBEX
TCS
Triggers

Index	Sl/Mast	Type	Freq	Description	Payload Data
11217	Master	DM1	2478 MHz	LMP_features_req	4E 6F F8 71 00 00 00 00 00
11220	Slave	DM1	2449 MHz	LMP_features_res	50 6F F8 71 00 00 00 00 00
11223	Master	DM1	2452 MHz	LMP_version_req	4A 01 0C 00 CD 00
11226	Slave	DM1	2453 MHz	LMP_version_res	4C 01 0C 00 CD 00
11229	Master	DM1	2468 MHz	LMP_host_connection_req	66
11231	Master	DM1	2405 MHz	LMP_host_connection_req	66
11590	Slave	DM1	2457 MHz	LMP_accepted	06 33
11593	Master	DM1	2425 MHz	LMP_setup_complete	62
11606	Slave	DM1	2469 MHz	LMP_setup_complete	63
11609	Master	DM1	2461 MHz	LMP_max_slot_req	5C 05
11612	Slave	DM1	2456 MHz	LMP_accepted	06 2E
11615	Master	DM1	2479 MHz	LMP_max_slot	5A 05
11629	Master	DM1	2470 MHz	LMP_clkoffset_req	0A
11632	Slave	DM1	2459 MHz	LMP_clkoffset_res	0C FF 1C
11635	Master	DM1	2423 MHz	LMP_features_req	4E 6F F8 71 00 00 00 00 00
11638	Slave	DM1	2461 MHz	LMP_features_res	50 6F F8 71 00 00 00 00 00
11641	Master	DM1	2407 MHz	LMP_version_req	4A 01 0C 00 CD 00
11644	Slave	DM1	2467 MHz	LMP_version_res	4C 01 0C 00 CD 00
11870	Slave	DM1	2442 MHz	LMP_max_slot_req	5D 05
11873	Master	DM1	2449 MHz	LMP_accepted	07 2E
11900	Slave	DM1	2425 MHz	LMP_features_req	4F 6F F8 71 00 00 00 00 00
11903	Master	DM1	2456 MHz	LMP_features_res	51 6F F8 71 00 00 00 00 00
11926	Slave	DM1	2432 MHz	LMP_version_req	4B 01 0C 00 CD 00
11929	Master	DM1	2431 MHz	LMP_version_res	4D 01 0C 00 CD 00

PDU OpCode	TID	features
0x27	0	0x6FF8710000000000

PDU	0x27 LMP_features_req
Transaction ID	0
features	0x6FF8710000000000

Estimated Clock	27231712
Hop Frequency	2478 MHz
Status	0x80: Packet Transmitted

0x27 - LMP_features_req

The Master requests the Slave for supported features. The Slave must reply with LMP_features_res.

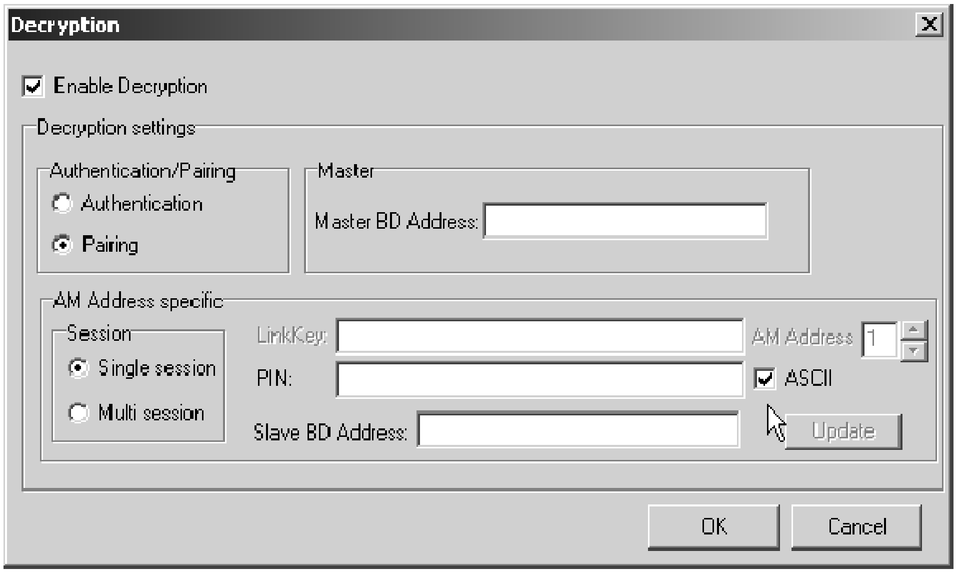
Features:
+ 3-slot packets
+ 5-slot packets
+ encryption
+ slot offsets
- timing accuracy
+ switch
+ hold mode
- sniff mode
- park mode

BaseBand: 12078 | LMP: 24 | L2CAP: 2 | RFCOMM: 0 | SDP: 0 | OBEX: 0 | TCS: 0 | Trigger: 1 | Filter: Off

Figure 2-15: Packet Analyzer display of error generated by the Data Collector

Decryption

Click this button to display the Decryption window. You use this window to enable decryption and enter settings. The procedure follows on the next page. See Figure 2-16 and Figure 2-17.



Decryption

☑ Enable Decryption

Decryption settings

Authentication/Pairing

○ Authentication

● Pairing

Master

Master BD Address:

AM Address specific

Session

● Single session

○ Multi session

LinkKey:

PIN:

Slave BD Address:

AM Address 1

☑ ASCII

Figure 2-16: Decryption window

The Data Collector is responsible for detection of Kc' (see Bluetooth Specification 1.0B or 1.1). Selecting this option is similar to the selection of piconet member mode in that the LinkKey and PIN (code) are requested through a dialog box.

When a log session is started, data is logged to the log file with packets for both encrypted and decrypted packets. The log file also includes LinkKey or PIN information.

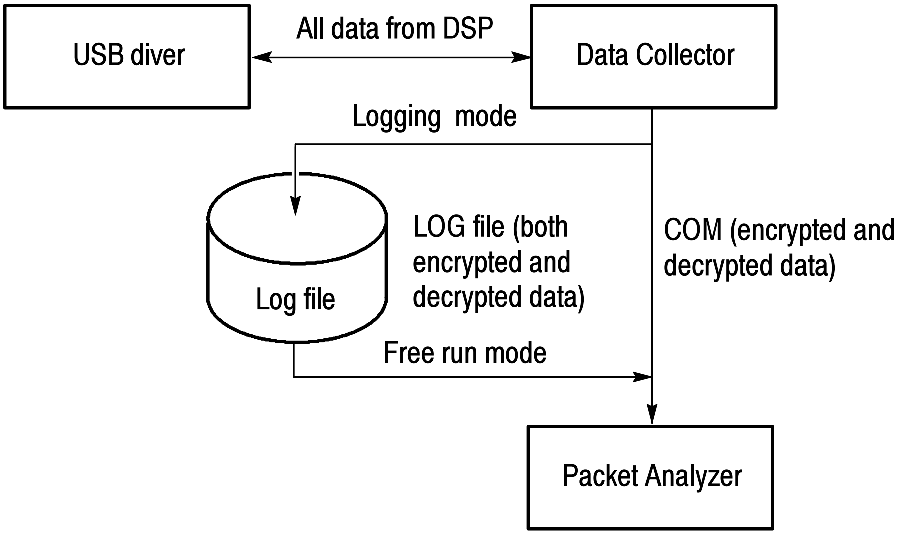


Figure 2-17: Decryption of Data diagram

The Packet Analyzer displays decrypted data in real time mode if performance is critical, or it can open a log file and display either decrypted or encrypted packets. In the case of encrypted packets, it is possible to decrypt using the LinkKey or PIN used during acquisition, or enter a LinkKey or PIN using the Decryption dialog box. This is explained in the procedure that follows on page 2-34.

Decryption in Independent Mode. Bluetooth security supports authentication (unidirectional or mutual) and encryption, which are based on a secret LinkKey that is shared by a pair of devices. This secret key is derived during initialization and is not disclosed.

Authentication. The size of the LinkKey is always 128 bit. In encryption it may vary from 8-128 bits (the authentication key is used in generating the encryption key).

Pairing. This is an authentication process. You do not have to calculate the LinkKey using a complex algorithm. Enter the PIN code (optional ASCII entry) used between master and slave for authentication. In pairing, the K_{init} value is calculated and used for decrypting the data transaction between master and slave (see note).

***NOTE**. When using decryption in Independent Mode with the Pairing option, there are some keys generated that are displayed in the Data Collector main window. The keys that are displayed are: Random number, Kc, Kc prime, and LinkKey. See Figure 2-4.*

Encryption Setup. The following setup has to be made in Bluetooth Neighborhood to enable encryption.

1. In the Bluetooth menu, select Bluetooth Neighborhood Properties-Security tab. For Security Mode, select Link level security and enable the Encryption Mode option.
2. Once bonding is established between master and slave, to use decryption in independent mode, you need to expire bonding. Right-click the device bonded in Bluetooth Neighborhood and select expire bonding.

Enable Encryption Procedure. Use the following procedure to enable decryption in the Data Collector:

1. From the Data Collector main window, select Decryption from the Filter menu.
2. In the Decryption dialog box (see Figure 2-16), click the Enable Decryption box.
3. Make your other selections from the following:
   - Authentication/Pairing. Choose either Authentication (default) or Pairing and follow these guidelines:
     - If using Authentication, enter the LinkKey.
     - If using Pairing, enter the PIN. The BPA100 Protocol Analyzer derives the LinkKey from the PIN. If entering the PIN in ASCII, click the ASCII check box as shown in Figure 2-16.
   - Master. Enter the Master BD Address.
   - AM Address specific. Choose Single session (default) or Multi session.
   - LinkKey/PIN. See Authentication/Pairing above.

- AM Address. Make selection.
- Slave BD Address. Enter the address.

4. Click OK. Example shows the Enable Decryption box checked.

HCI Terminal

The HCI Terminal application provides a hardware interface similar to the interface provided by an AT terminal application when communicating with a modem. The HCI Terminal application provides control of the BPA100 in piconet member mode. This is similar to using the Bluetooth Neighborhood from the Software Suite.

How to create HCI scripts. The *HCI Terminal Guide* describes the functionality of the script language. The sample scripts provided will help you to understand HCI scripting.

***NOTE**. The HCI Terminal application and Bluetooth Neighborhood cannot both be used at the same time. For error generation you are advised to use the HCI terminal instead of Bluetooth Neighborhood.*

Exiting the Data Collector

To exit a log session in the Data Collector, select Exit from the File menu.

Packet Analyzer Operation

The Bluetooth Packet Analyzer analyzes and displays the contents of the log files created by the Data Collector. The Bluetooth Packet Analyzer can do the following:

- Analyze and decode packet information at Baseband, LMP, L2CAP, RFCOMM, SDP, OBEX, and TCS protocol levels
- Export data to .CSV (comma separated value) files readable by other applications, such as Microsoft Excel
- Display error packets and access errors
- Indicate trigger packets, defined sequences, and generate error packets
- Display packets continuously as the packets are received and logged (this free run mode is initiated in the Bluetooth Data Collector application, see Figure 2-2 on page 2-6.)

Main Window

Figure 2-18 shows the main window of the Bluetooth Packet Analyzer. Each of the major areas of the main window is described in the text associated with the number of the area.

1. Menu bar. The menu bar contains the File, Edit, View, and Help menus and their associated menu items.

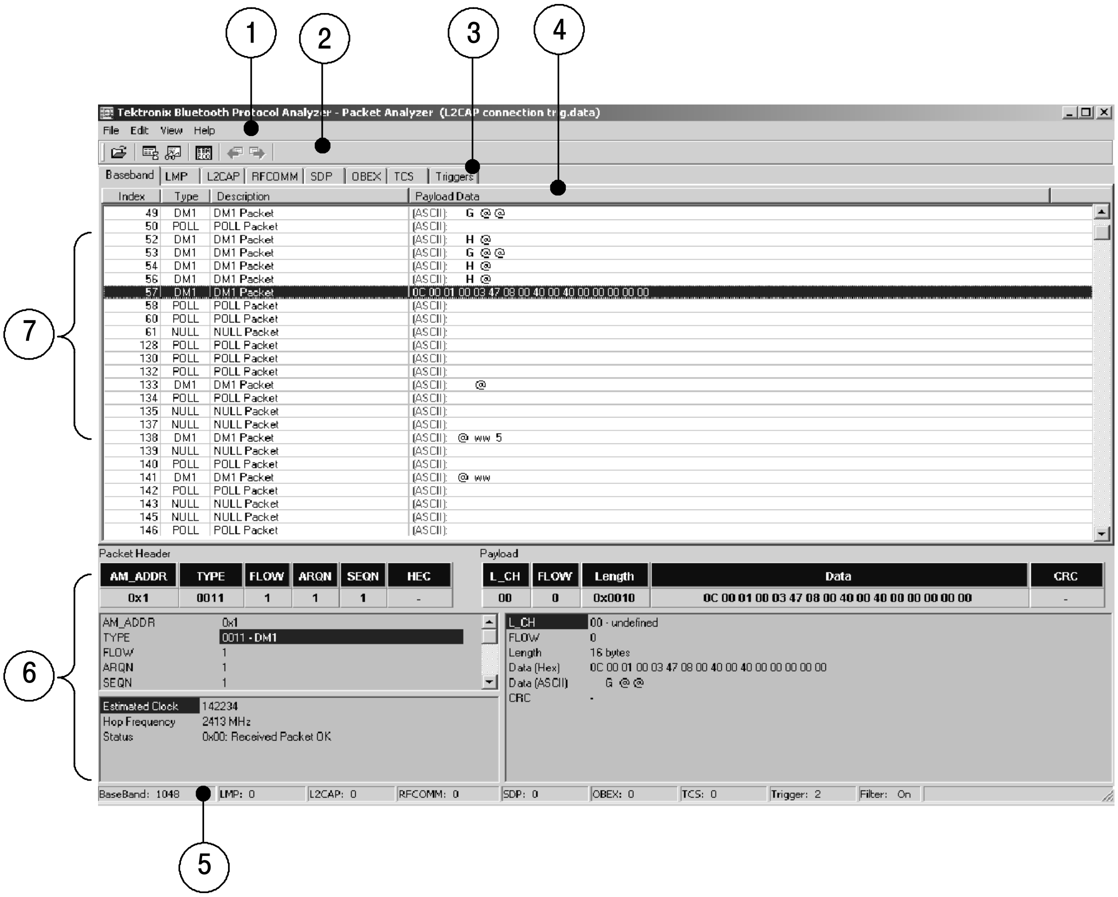
   From the File menu you can:

   - Open files (see *Opening a File* on page 2-39)
   - Export data to a comma separated value file (.csv)
   - View the properties of the current log file, such as Name and Size
   - Exit the application

   From the Edit menu you can:

   - Switch a bookmark on or off (see *Toggle Bookmark* on page 2-44)

- Set the L2CAP connection properties (see *L2CAP Connection Properties* on page 2-45)
- Highlight L2CAP connections (see *Highlight L2CAP Connection* on page 2-45)


1
2
3
4
5
6
7
Tektronix Bluetooth Protocol Analyzer - Packet Analyzer (L2CAP connection trig.data)
File
Edit
View
Help
Baseband
LMP
L2CAP
RFCOMM
SDP
OBEX
TCS
Triggers

Index	Type	Description	Payload Data
49	DM1	DM1 Packet	(ASCII): G @ @
50	POLL	POLL Packet	(ASCII):
52	DM1	DM1 Packet	(ASCII): H @
53	DM1	DM1 Packet	(ASCII): G @ @
54	DM1	DM1 Packet	(ASCII): H @
56	DM1	DM1 Packet	(ASCII): H @
57	DM1	DM1 Packet	0C 00 01 00 03 47 08 00 40 00 40 00 00 00 00 00
58	POLL	POLL Packet	(ASCII):
60	POLL	POLL Packet	(ASCII):
61	NULL	NULL Packet	(ASCII):
128	POLL	POLL Packet	(ASCII):
130	POLL	POLL Packet	(ASCII):
132	POLL	POLL Packet	(ASCII):
133	DM1	DM1 Packet	(ASCII): @
134	POLL	POLL Packet	(ASCII):
135	NULL	NULL Packet	(ASCII):
137	NULL	NULL Packet	(ASCII):
138	DM1	DM1 Packet	(ASCII): @ ww 5
139	NULL	NULL Packet	(ASCII):
140	POLL	POLL Packet	(ASCII):
141	DM1	DM1 Packet	(ASCII): @ ww
142	POLL	POLL Packet	(ASCII):
143	NULL	NULL Packet	(ASCII):
145	NULL	NULL Packet	(ASCII):
146	POLL	POLL Packet	(ASCII):

Packet Header

AM_ADDR	TYPE	FLOW	ARQN	SEQN	HEC
0x1	0011	1	1	1	-

Payload

L_CH	FLOW	Length	Data	CRC
00	0	0x0010	0C 00 01 00 03 47 08 00 40 00 40 00 00 00 00 00	-

AM_ADDR	0x1
TYPE	0011 - DM1
FLOW	1
ARQN	1
SEQN	1

Estimated Clock	142234
Hop Frequency	2413 MHz
Status	0x00: Received Packet OK

L_CH	00 - undefined
FLOW	0
Length	16 bytes
Data (Hex)	0C 00 01 00 03 47 08 00 40 00 40 00 00 00 00 00
Data (ASCII)	G @ @
CRC	-

BaseBand: 1048 | LMP: 0 | L2CAP: 0 | RFCOMM: 0 | SDP: 0 | OBEX: 0 | TCS: 0 | Trigger: 2 | Filter: On

Figure 2-18: Main window of the Bluetooth Packet Analyzer

- Highlight AM_ADDR (see *Highlight AM_ADDR* on page 2-45)
- Highlight fragmentation (see *Highlight Fragmentation* on page 2-45)

- Switch the display of payload data between Hexadecimal or ASCII format (see *Toggle Hex/ASCII in Payload* on page 2-45)
- Clear toggled fields (see *Clear Toggled Fields* on page 2-45)
- Clear highlights (see *Clear Highlights* on page 2-45)

From the View menu, you can:

- Switch the toolbar on or off
- View and change the filter setup (see *Filter Setup* on page 2-40)
- View and change the view setup (see *View Setup* on page 2-41)
- Open the Bookmarks window (see Figure 2-23 on page 2-44)
- Go directly to any packet number that you want
- Search (Find and Find Next)
- Open and change the Packet Hex View window (see *Hex View* on page 2-42).
- Switch the display of packet information on or off

From the Help menu you can:

- Connect to the Tektronix web site
- Connect to the Digianswer web site
- See information about the Bluetooth Packet Analyzer, such as the version number

2. Toolbar buttons. These buttons are shortcuts to many of the functions of the Bluetooth Packet Analyzer. Each of the buttons has a corresponding menu item in the menu bar (except for the Go One Level Back and Go to Next Level buttons). The buttons are described in *Packet Analyzer Toolbar Buttons* on page 2-39.

3. Tabs. Use these tabs to select which packets of the current log file you want to see–all baseband packets or specific types and levels of packets, such as LMP, L2CAP, RFCOMM, and SDP. The Triggers tab displays triggers and trigger-arming events that you have defined. The OBEX tab displays file-transfer and business card data. The TCS tab displays protocol discriminator, message type, and other data (depending upon the message type).

4. Columns. These columns reflect the elements that you selected in the view setup, where you can decide which elements you want the list view to show. See *View Setup* on page 2–41 for more information.

5. Status bar. The status bar displays the number of packets logged of the type: Baseband, LMP, L2CAP, RFCOMM, SDP, OBEX, and TCS. It also displays the number of trigger packets and indicates whether a filter (see *Filter Setup* on page 2–40) is selected for the packet type being displayed.

6. Packet data. This area displays information about the packet currently highlighted in the list view. The type of information that is displayed depends on the type and contents of the packet. Various elements (columns) of packet data can be switched off or on in the View Setup dialog box (see Figure 2–20 on page 2–41).

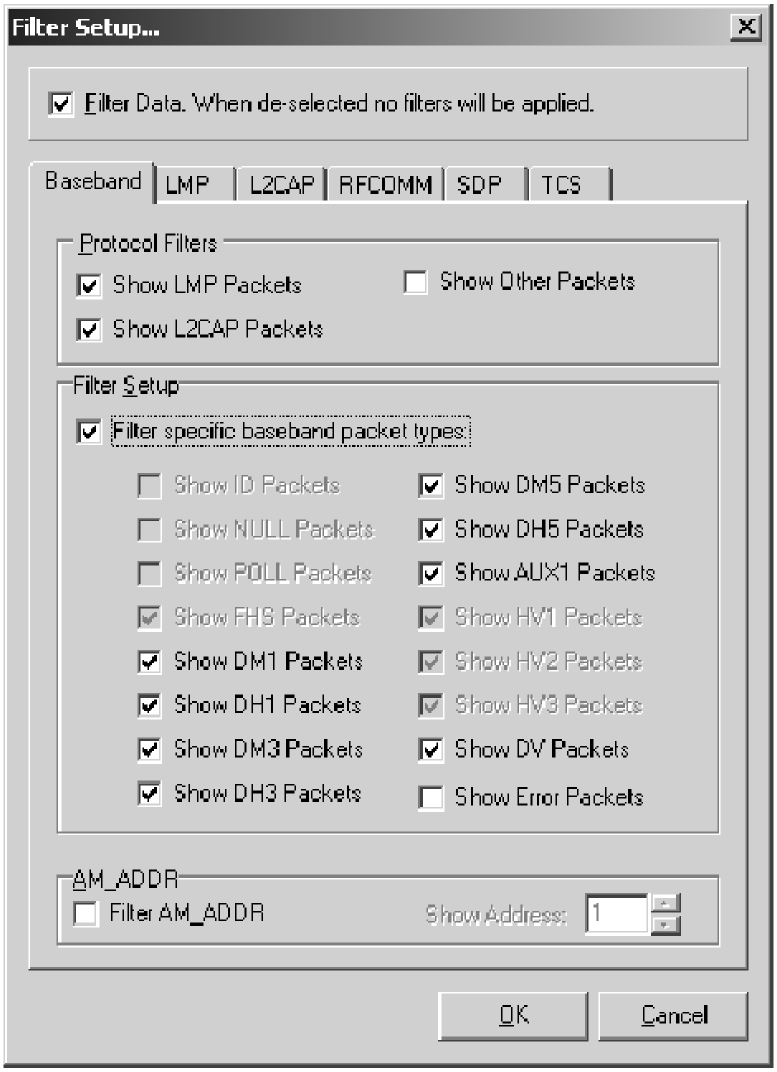
7. List view. The list view displays the contents of the current log file as a list of the packets that the file contains. If the Data Collector is set to free run mode, the list view will display packet data as it is received and logged. You can start and stop the automatic screen updates by pressing the Esc key on your keyboard.

Packet Analyzer Toolbar Buttons

The following are descriptions of the toolbar buttons available for the Bluetooth Packet Analyzer:

Opening a File. Click this button to display the Open dialog box that allows you to browse and open log files that have the .data extension. The Protocol Analyzer features especially fast load of files up to the available physical and virtual memory limitations. Files exceeding this size will be loaded at a slower rate.

Filter Setup. Click this button to open the Filter Setup... dialog box (see Figure 2-19). The filter function allows you to reduce the amount of data displayed in the list view. In Baseband, for example, you can choose to view only LMP and L2CAP packets, rather than all of the transmitted packets. This function can greatly reduce the number of packets in a log session, making it much easier for you to work with the data.



☑ Filter Data. When de-selected no filters will be applied.

Protocol Filters

☑ Show LMP Packets

☑ Show L2CAP Packets

□ Show Other Packets

Filter Setup

☑ Filter specific baseband packet types

□ Show ID Packets
□ Show NULL Packets
□ Show POLL Packets
☑ Show FHS Packets
☑ Show DM1 Packets
☑ Show DH1 Packets
☑ Show DM3 Packets
☑ Show DH3 Packets

☑ Show DM5 Packets
☑ Show DH5 Packets
☑ Show AUX1 Packets
☑ Show HV1 Packets
☑ Show HV2 Packets
☑ Show HV3 Packets
☑ Show DV Packets
□ Show Error Packets

AM_ADDR

□ Filter AM_ADDR

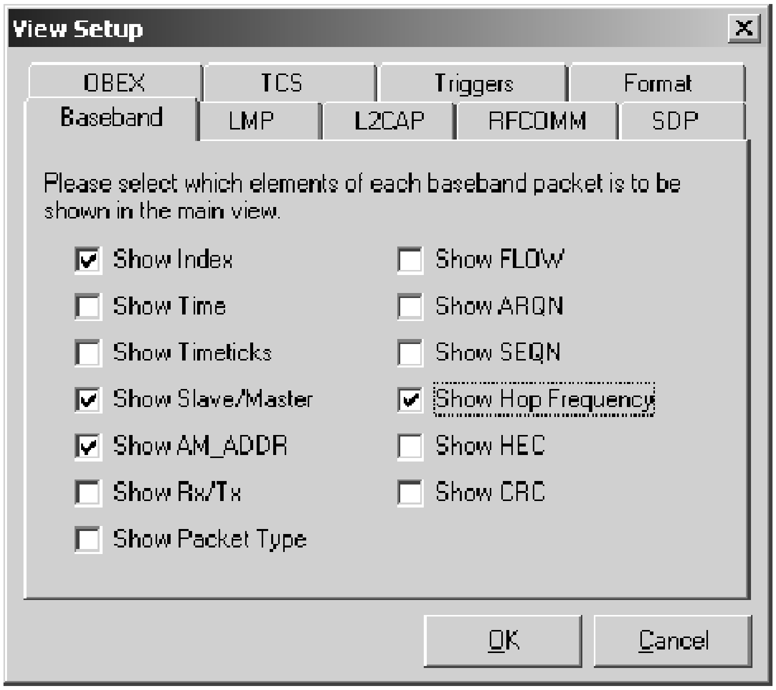
Show Address: 1

Figure 2-19: Filter Setup dialog box

The filter function can be selected for each of the Baseband, LMP, L2CAP, RFCOMM, TCS, or SDP lists; applying a filter to one of these lists does not affect the others.

In the Filter Setup dialog box, you can select the list of packets to which the filter function is applied and which data is filtered. The choice of data varies, depending on which list of packets is selected. Figure 2-19 shows the dialog box as it appears when Baseband is selected.

View Setup. Click this button to open the View Setup dialog box (see Figure 2-20). For each of the tabs representing a packet type (Baseband, LMP, L2CAP, RFCOMM, SDP, OBEX, and TCS), you can select which elements are displayed in the list view. You can click the Triggers tab to view triggers that you have set up. You can also click the Format tab to change the display radix or type (for example, decimal, hexadecimal, or binary).



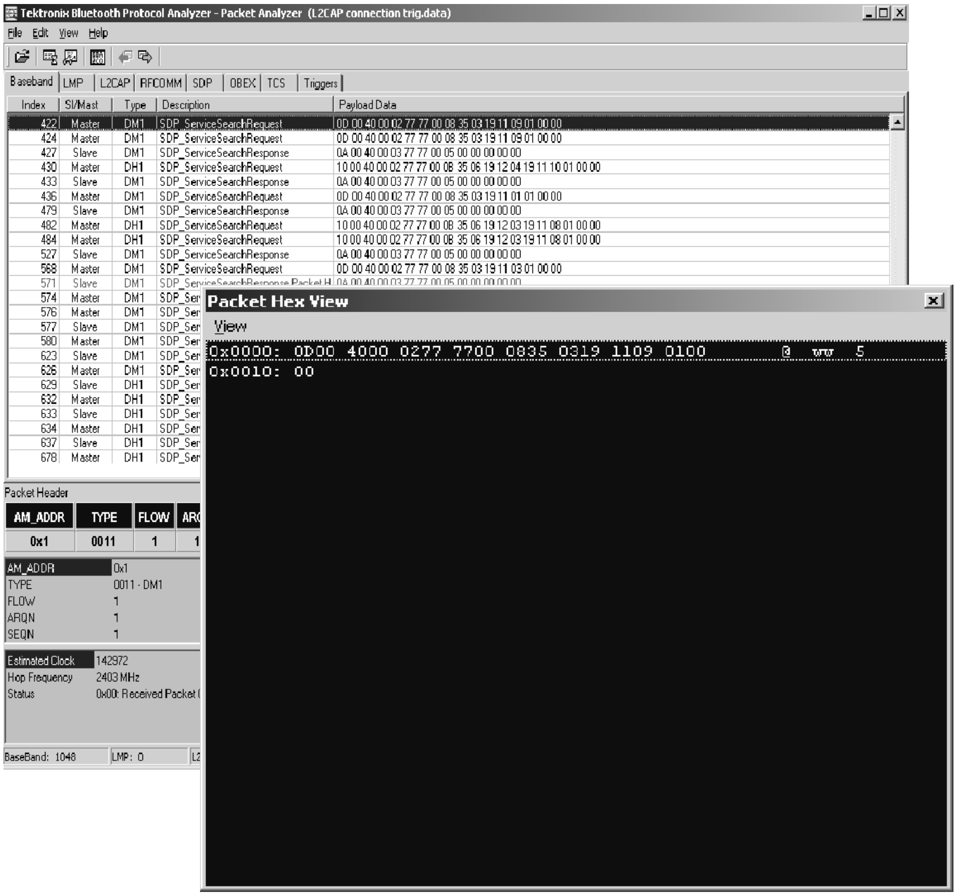
Please select which elements of each baseband packet is to be shown in the main view.

☑ Show Index
☐ Show Time
☐ Show Timeticks
☑ Show Slave/Master
☑ Show AM_ADDR
☐ Show Rx/Tx
☐ Show Packet Type
☐ Show FLOW
☐ Show ARQN
☐ Show SEQN
☑ Show Hop Frequency
☐ Show HEC
☐ Show CRC

Figure 2-20: View Setup dialog box

In Figure 2-20, the following elements have been selected for Baseband: Index, Slave/Master, AM_ADDR, and Hop Frequency. By default, the Description and Payload Data tabs are always present in the Packet Analyzer main window.

Hex View. Click this button to open the Packet Hex View window (see Figure 2-21).


Tektronix Bluetooth Protocol Analyzer - Packet Analyzer (L2CAP connection trig.data)
File
Edit
View
Help
Baseband
LMP
L2CAP
RFCOMM
SDP
OBEX
TCS
Triggers
Packet Hex View
View

Index	Sl/Mast	Type	Description	Payload Data
422	Master	DM1	SDP_ServiceSearchRequest	0D 00 40 00 02 77 77 00 08 35 03 19 11 09 01 00 00
424	Master	DM1	SDP_ServiceSearchRequest	0D 00 40 00 02 77 77 00 08 35 03 19 11 09 01 00 00
427	Slave	DM1	SDP_ServiceSearchResponse	0A 00 40 00 03 77 77 00 05 00 00 00 00 00
430	Master	DH1	SDP_ServiceSearchRequest	10 00 40 00 02 77 77 00 08 35 06 19 12 04 19 11 10 01 00 00
433	Slave	DM1	SDP_ServiceSearchResponse	0A 00 40 00 03 77 77 00 05 00 00 00 00 00
436	Master	DM1	SDP_ServiceSearchRequest	0D 00 40 00 02 77 77 00 08 35 03 19 11 01 01 00 00
479	Slave	DM1	SDP_ServiceSearchResponse	0A 00 40 00 03 77 77 00 05 00 00 00 00 00
482	Master	DH1	SDP_ServiceSearchRequest	10 00 40 00 02 77 77 00 0B 35 06 19 12 03 19 11 08 01 00 00
484	Master	DH1	SDP_ServiceSearchRequest	10 00 40 00 02 77 77 00 0B 35 06 19 12 03 19 11 08 01 00 00
527	Slave	DM1	SDP_ServiceSearchResponse	0A 00 40 00 03 77 77 00 05 00 00 00 00 00
568	Master	DM1	SDP_ServiceSearchRequest	0D 00 40 00 02 77 77 00 08 35 03 19 11 03 01 00 00
571	Slave	DM1	SDP_ServiceSearchResponse Packet H	0A 00 40 00 03 77 77 00 05 00 00 00 00 00
574	Master	DM1	SDP_Ser	
576	Master	DM1	SDP_Ser	
577	Slave	DM1	SDP_Ser	
580	Master	DM1	SDP_Ser	
623	Slave	DM1	SDP_Ser	
626	Master	DM1	SDP_Ser	
629	Slave	DH1	SDP_Ser	
632	Master	DH1	SDP_Ser	
633	Slave	DH1	SDP_Ser	
634	Master	DH1	SDP_Ser	
637	Slave	DH1	SDP_Ser	
678	Master	DH1	SDP_Ser	

Packet Header

AM_ADDR	TYPE	FLOW	ARC
0x1	0011	1	1

AM_ADDR	0x1
TYPE	0011 - DM1
FLOW	1
ARQN	1
SEQN	1

Estimated Clock	142972
Hop Frequency	2403 MHz
Status	0x00: Received Packet (

BaseBand: 1048 | LMP: 0 | L2

```
0x0000:  0D00  4000  0277  7700  0835  0319  1109  0100        @  ww  5
0x0010:  00
```

Figure 2-21: Packet Hex View window displayed on top of Main window

The main window only shows the first several bytes of what a packet contains. However, you can view the entire contents of a packet of any length by opening the Packet Hex View window. In the View menu of this window, you can select Hex or Binary. Also, you can select Stay On Top to keep the dialog box in front of any other Bluetooth Packet Analyzer windows that are open.

Go One Level Back. Click this button to go to the next lower protocol (lowest level is baseband) for the packet highlighted in the list view. You can also press the Backspace key to move to the next lower protocol.

Go to Next Level. Click this button to go to the next higher level of protocol for the packet highlighted in the list view. You can also press the Enter key to go to the next higher level of protocol.

***NOTE**. Clicking the tabs will change levels of protocol but will not maintain highlighting or necessarily display the same packet.*

Context Menu

You can right-click in the list view area of the main window (see Figure 2-18 on page 2-37) to display the context-sensitive menu shown in Figure 2-22.

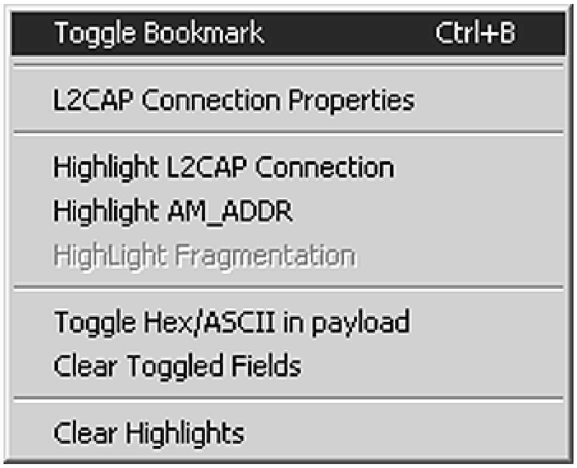


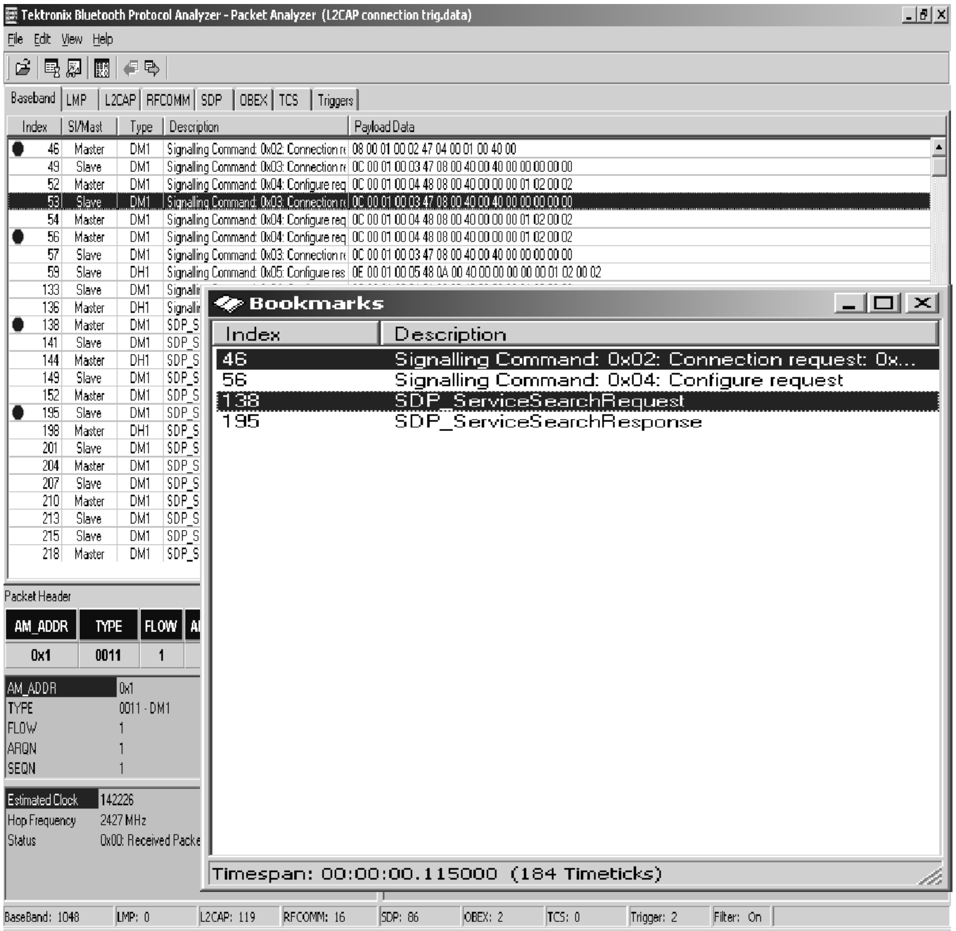
Figure 2-22: List view context-sensitive menu

The menu items in the context-sensitive menu are discussed on the following page.

Toggle Bookmark. Click this menu item to toggle a bookmark on or off for the packet that you have highlighted in the list view. When a bookmark is assigned to a packet, a large, blue bullet is placed at the left side of the Index field for the highlighted packet. See Figure 2-23.

Bookmarks allow you to quickly display packets in which you are interested. To move to a bookmarked packet, go to the View menu in the Menu bar, and select Bookmarks. The Bookmarks dialog box is displayed. See Figure 2-23.

Double-click the bookmarked packet that you want to display in the list view.


Tektronix Bluetooth Protocol Analyzer - Packet Analyzer (L2CAP connection trig.data)
File
Edit
View
Help
Baseband
LMP
L2CAP
RFCOMM
SDP
OBEX
TCS
Triggers
Bookmarks

Index	Sl/Mast	Type	Description	Payload Data
● 46	Master	DM1	Signalling Command: 0x02: Connection re	08 00 01 00 02 47 04 00 01 00 40 00
49	Slave	DM1	Signalling Command: 0x03: Connection re	0C 00 01 00 03 47 08 00 40 00 40 00 00 00 00 00
52	Master	DM1	Signalling Command: 0x04: Configure req	0C 00 01 00 04 48 08 00 40 00 00 00 01 02 00 02
53	Slave	DM1	Signalling Command: 0x03: Connection re	0C 00 01 00 03 47 08 00 40 00 40 00 00 00 00 00
54	Master	DM1	Signalling Command: 0x04: Configure req	0C 00 01 00 04 48 08 00 40 00 00 00 01 02 00 02
● 56	Master	DM1	Signalling Command: 0x04: Configure req	0C 00 01 00 04 48 08 00 40 00 00 00 01 02 00 02
57	Slave	DM1	Signalling Command: 0x03: Connection re	0C 00 01 00 03 47 08 00 40 00 40 00 00 00 00 00
59	Slave	DH1	Signalling Command: 0x05: Configure res	0E 00 01 00 05 48 0A 00 40 00 00 00 00 00 01 02 00 02
133	Slave	DM1	Signallir	
136	Master	DH1	Signallir	
● 138	Master	DM1	SDP_S	
141	Slave	DM1	SDP_S	
144	Master	DH1	SDP_S	
149	Slave	DM1	SDP_S	
152	Master	DM1	SDP_S	
● 195	Slave	DM1	SDP_S	
198	Master	DH1	SDP_S	
201	Slave	DM1	SDP_S	
204	Master	DM1	SDP_S	
207	Slave	DM1	SDP_S	
210	Master	DM1	SDP_S	
213	Slave	DM1	SDP_S	
215	Slave	DM1	SDP_S	
218	Master	DM1	SDP_S	

Index	Description
46	Signalling Command: 0x02: Connection request: 0x...
56	Signalling Command: 0x04: Configure request
138	SDP_ServiceSearchRequest
195	SDP_ServiceSearchResponse

Timespan: 00:00:00.115000 (184 Timeticks)

Packet Header

AM_ADDR	TYPE	FLOW	AI
0x1	0011	1	

AM_ADDR	0x1
TYPE	0011 - DM1
FLOW	1
ARQN	1
SEQN	1

Estimated Clock	142226
Hop Frequency	2427 MHz
Status	0x00: Received Packe

BaseBand: 1048 | LMP: 0 | L2CAP: 119 | RFCOMM: 16 | SDP: 86 | OBEX: 2 | TCS: 0 | Trigger: 2 | Filter: On

Figure 2-23: Bookmarks dialog box

You can measure the time between any two bookmarks in the Bookmarks dialog box. First click one of the bookmarks to select it. Then control-click the other bookmark to highlight it. Read the time between the bookmarks at the bottom of the Bookmarks dialog box (see Figure 2-23) displayed in hours, minutes, seconds, and microseconds. Also, time is given in timeticks (625 μs per timetick).

L2CAP Connection Properties. Click this menu item to set the L2CAP packet type for acquisitions where the packet type cannot be decoded from previous packets.

Highlight L2CAP Connection. Click this menu item to highlight the L2CAP packets in Baseband.

Highlight AM_ADDR. Click this menu item to highlight the AM_ADDR of active slaves that are connected to the master. AM_ADDR ranges from 0 through 7.

Highlight Fragmentation. Click this menu item when you have a fragmented packet selected to highlight all the fragmented packets. Fragmentation occurs when the payload data is large enough that it must be segmented and transmitted with more than one packet.

Toggle Hex/ASCII in Payload. Click this menu item to switch the display of the payload data for the highlighted packet between hexadecimal format and ASCII format.

Clear Toggled Fields. Click this item to return fields that you changed with a toggle to their original format (does not affect bookmarks).

Clear Highlights. Click this menu item to clear any highlights that you have set, such as highlights for the AM_ADDR.

Exporting Data

1. On the menu bar click File, and then select Export.

2. In the Export dialog box, select a path/folder, and name the file.

3. Click OK.

Exiting the Bluetooth Packet Analyzer

- On the menu bar click File, and then select Exit.

Reference

Reference

This section provides technical information that you may need, such as hardware specifications and Bluetooth radio specifications.

Hardware Specifications

The Bluetooth hardware specifications are as follows:

- Compliant with the USB Specification, Version 1.1
- Powered through USB cable connected between the host PC and the Bluetooth Air Interface Probe
- Standby power consumption is 81 mA when Inquiry scan is enabled; 400 μA in Hibernation or Standby mode
- Active power consumption is less than 350 mA

Bluetooth Radio Specifications

The radio specifications for the Bluetooth Air Probe are as follows:

- Bluetooth-qualified device
- Transmit power is +20 dBm (100 mW) in normal mode; 0 dBm, 1 mW in Single Frequency mode
- Receiver sensitivity is better than -80 dBm
- Frequency range is 2.402 - 2.480 GHz

Environmental Specifications

The environmental specifications for the Bluetooth Air Interface Probe are as follows:

- Temperature, operating: 41 °F to 122 °F (+5°C to 50 °C)
- Temperature, nonoperating: -4 °F to 140 °F (-20 °C to +60 °C)
- Humidity: 20% to 80%
- Altitude, operating: 1000 ft to 10,000 ft (305 m to 3,050 m)
- Range: 0 ft to 820 ft (0-250 m)

Dimensions of the Bluetooth Air Probe

Figure 3-1 shows the dimensions of the Bluetooth Air Probe.

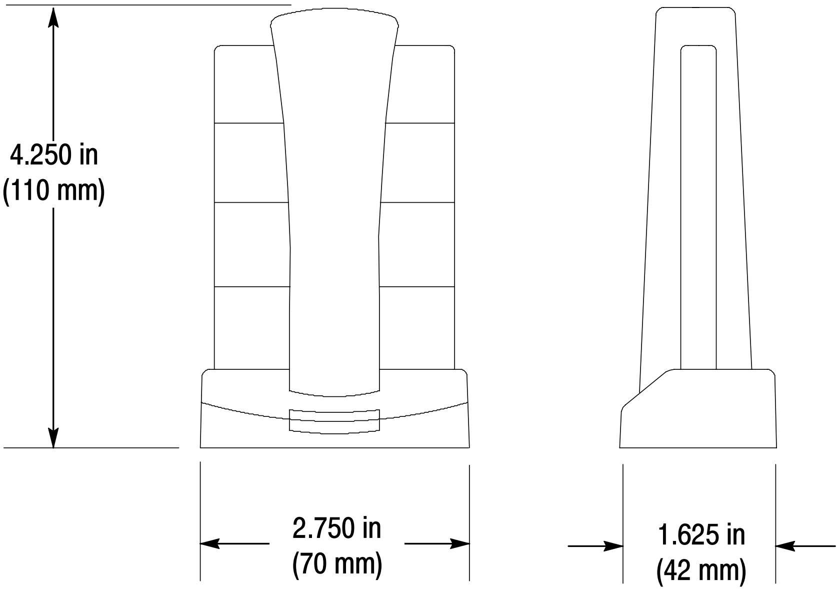


Figure 3-1: Dimensions of the Bluetooth Air Probe

HCI Terminal Sample Scripts

Use the following samples as a guide to create your scripts.

Sniffer testscript for Master packet types.

```
report(Sniffer testscript for packet types [Master])
report( )

RESET(All)
SETDEBUGLEVEL(81)
SETMAXLOOPCOUNT(5000)
WAITCOMPLETE_ENABLED
//TIMESTAMPS_ENABLED

// Write Scan enable
// Set Event Filter
// Change connection packet type

TXCMD 1A 0C 01 00
WAITEVENT($0E,5000,[TestError])
TXCMD 05 0C 03 02 00 02
WAITEVENT($0E,5000,[TestError])

// Establish ACL connection

report( )
report(Establishing ACL connection)

label: Establish_one_connection
label: create_connection_retry#1

// NOTE:
// change the Bluetooth address in this command
// if your BD_Addr is 00 50 CD 00 93 38 then it should be reversed
as 38 93 00 CD 50 00
// Its starts |              | it is reversed
TXCMD 05 04 0C 38 93 00 CD 50 00 18 CC 00 00 00 00
WAITEVENT($03,20000,[TestError])
if byte[2] = $04 jump(create_connection_retry#1)
if byte[2] = $10 jump(create_connection_retry#1)
```

```
report(ACL connection established!)
report( )

delay(1000)
//WAITEVENT($1B,5000,[TestError])
WAITEVENT($1C,5000,[TestError])
WAITEVENT($0B,5000,[TestError])
WAITEVENT($0C,5000,[TestError])

//TXCMD 0F 04 04 00 00 18 CC
//WAITEVENT($1D,5000,[TestError])

report(Connection packet type changed)
report( )

// switch from master to slave

TXCMD 0B 08 07 38 93 00 CD 50 00 00
WAITEVENT($12,1000,[TestError])

// Disconnect ACL connection
// This Device is Slave now so wait for Disconnect from master

label: Disconnect

//TXCMD 06 04 03 00 00 13
WAITEVENT($05,60000,[TestError])
report(ACL connection disconnected)
report( )

label: TestSuccess
report(Test passed!)
report( )
jump(end)

label: TestError

report()
report(***************Test failed!*******************)
report()
```

```
label: end
REPORT(DONE!)
```

Sniffer testscript for Slave packet types.

```
report(Sniffer testscript for packet types [Slave])
report( )

RESET(All)
SETDEBUGLEVEL(81)
SETMAXLOOPCOUNT(5000)
WAITCOMPLETE_ENABLED
//TIMESTAMPS_ENABLED

// Write Scan enable
// Set Event Filter
// Wait for max slots changed event

TXCMD 1A 0C 01 03
//WAITEVENT($0E,5000,[TestError])
TXCMD 05 0C 03 02 00 02
WAITEVENT($0E,5000,[TestError])

// Establish ACL connection

report( )
report(Establishing ACL connection)

WAITEVENT($03,60000,[TestError])

report(ACL connection established from master!)
report( )

delay(1000)

WAITEVENT($1B,60000,[TestError])
WAITEVENT($1C,60000,[TestError])
WAITEVENT($0B,60000,[TestError])
WAITEVENT($0C,60000,[TestError])
```

```
//WAITEVENT($1B,5000,[TestError])
//report(Connection packet type changed from master)
report( )

// ROLE Switch this device becomes master

//WAITEVENT($12,10000,[TestError])
delay(6000)

// Wait for master to disconnect ACL connection
// This device is master now so disconnect the connection

label: Disconnect

TXCMD 06 04 03 00 00 13
WAITEVENT($05,10000,[TestError])
report(ACL connection disconnected from master)
report( )

label: TestSuccess
report(Test passed!)
jump(end)

label: TestError
report()
report(****************Test failed!***************)
report()

label: end
REPORT(DONE!)
```

Sniffer testscript for Slave connection packet types.

```
report(BPA100 connection testscript for packet types [Slave])
report( )

RESET(All)
SETDEBUGLEVEL(81)
SETMAXLOOPCOUNT(5000)
WAITCOMPLETE_ENABLED
```

```
//TIMESTAMPS_ENABLED

// Write Scan enable
// Set Event Filter
// Wait for max slots changed event

TXCMD 1A 0C 01 03
WAITEVENT($0E,5000,[TestError])
TXCMD 05 0C 03 02 00 02
WAITEVENT($0E,5000,[TestError])

REPORT(The following tests are from the test specification)

// Wait for events from master
// When master is done add 1 SCO HV1 connection and disconnect it
5.5.18.1.4 & 5.5.18.1.10

// Establish ACL connection

report( )
report(Establishing ACL connection)

WAITEVENT($03,60000,[TestError])
report(ACL connection established from master!)
report( )

WAITEVENT($1B,5000,[TestError])
report(Connection packet type changed from master)
report( )

// Set some payload
SETPAYLOAD(49 66 20 79 6F 75 20 63 61 6E 20 72 65 61 64 20 74
68 69 73 20 74 68 65 6E 20 79 6F 75 20 68 61 76 65 20 73 65 74 20
74 68 65 20 66 6F 72 6D 61 74 20 6F 66 20 74 68 65 20 70 61 79 6C
6F 61 64 20 74 6F 20 62 65 20 64 69 73 70 6C 61 79 65 64 20 69 6E
20 41 53 43 49 49 2E 20 53 6F 6D 65 74 69 6D 65 73 20 74 68 65
20 50 43 20 67 75 79 73 20 66 6F 72 67 65 74 73 20 74 6F 20 77 72
61 70 20 74 68 65 20 70 61 79 6C 6F 61 64 20 73 6F 20 79 6F 75 20
63 61 6E 20 6E 6F 74 20 73 65 65 20 69 74 20 61 6C 6C 20 61 74
20 6F 6E 65 20 74 69 6D 65 20 74 68 65 6E 20 79 6F 75 20 77 69
6C 6C 20 68 61 76 65 20 74 6F 20 63 68 6F 73 65 20 48 45 58 20 76
69 65 77 20 74 6F 20 73 65 65 20 69 74 20 61 6C 6C 2E 20 49 20 74
68 69 6E 6B 20 74 68 69 73 20 73 68 6F 75 6C 64 20 62 65 20 63 68
61 6E 67 65 64 20 61 73 20 73 6F 6F 6E 20 61 73 20 70 6F 73 73 69
62 6C 65 2C 20 68 6F 77 65 76 65 72 20 69 66 20 79 6F 75 20 63 61
6E 20 72 65 61 64 20 74 68 69 73 20 6C 69 6E 65 20 74 68 65 20 70
72 6F 62 6C 65 6D 20 69 73 20 66 69 78 65 64 20 21)

// Test DM1, DH1, DM3, DH3, DM5, DH5 packets

label: NoSCO

REPORT(Testing for DM1, DH1, DM3, DH3, DM5, DH5 packets)
report( )

TXDATA(hCon:0,bc:0,pb:2,Len:1,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:2,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:3,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:4,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:5,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:6,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:7,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:8,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:9,cnt:500,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:10,cnt:500,Random:0)

report()
report(Packets size = 1..10 ”passed”)
report()

// Wait for master to disconnect ACL connection

WAITEVENT($05,60000,[TestError])
report(ACL connection disconnected from master)
report( )

label: TestSuccess
report(Test passed!)
jump(end)
```

```
label: TestError
report(Test failed!)
label: end
REPORT(DONE!)
```

Sniffer testscript for Master connection packet types.

```
report(BPA100 Connection testscript [Master])
report( )

RESET(All)
SETDEBUGLEVEL(81)
SETMAXLOOPCOUNT(5000)
WAITCOMPLETE_ENABLED
//TIMESTAMPS_ENABLED

// Write Scan enable
// Set Event Filter
// Change connection packet type

TXCMD 1A 0C 01 00
WAITEVENT($0E,5000,[TestError])
TXCMD 05 0C 03 02 00 02
WAITEVENT($0E,5000,[TestError])

// Establish ACL connection

report( )
report(Establishing ACL connection)

label: Establish_one_connection
label: create_connection_retry#1

// NOTE:
// change the Bluetooth address in this command
// if you BD_Addr is 00 50 CD 00 93 11 then it should be reversed as
11 93 00 CD 50 00
// Its starts |               | it is reversed
TXCMD 05 04 0C 11 93 00 CD 50 00 18 CC 00 00 00 00
WAITEVENT($03,20000,[TestError])
if byte[2] = $04 jump(create_connection_retry#1)
if byte[2] = $10 jump(create_connection_retry#1)

report(ACL connection established!)
report( )

//TXCMD 0F 04 04 00 00 18 CC
//WAITEVENT($1D,5000,[TestError])

report(Connection packet type changed)
report( )

// Set some payload
SETPAYLOAD(49 66 20 79 6F 75 20 63 61 6E 20 72 65 61 64 20 74
68 69 73 20 74 68 65 6E 20 79 6F 75 20 68 61 76 65 20 73 65 74 20
74 68 65 20 66 6F 72 6D 61 74 20 6F 66 20 74 68 65 20 70 61 79 6C
6F 61 64 20 74 6F 20 62 65 20 64 69 73 70 6C 61 79 65 64 20 69 6E
20 41 53 43 49 49 2E 20 53 6F 6D 65 74 69 6D 65 73 20 74 68 65
20 50 43 20 67 75 79 73 20 66 6F 72 67 65 74 73 20 74 6F 20 77 72
61 70 20 74 68 65 20 70 61 79 6C 6F 61 64 20 73 6F 20 79 6F 75 20
63 61 6E 20 6E 6F 74 20 73 65 65 20 69 74 20 61 6C 6C 20 61 74
20 6F 6E 65 20 74 69 6D 65 20 74 68 65 6E 20 79 6F 75 20 77 69
6C 6C 20 68 61 76 65 20 74 6F 20 63 68 6F 73 65 20 48 45 58 20 76
69 65 77 20 74 6F 20 73 65 65 20 69 74 20 61 6C 6C 2E 20 49 20 74
68 69 6E 6B 20 74 68 69 73 20 73 68 6F 75 6C 64 20 62 65 20 63 68
61 6E 67 65 64 20 61 73 20 73 6F 6F 6E 20 61 73 20 70 6F 73 73 69
62 6C 65 2C 20 68 6F 77 65 76 65 72 20 69 66 20 79 6F 75 20 63 61
6E 20 72 65 61 64 20 74 68 69 73 20 6C 69 6E 65 20 74 68 65 20 70
72 6F 62 6C 65 6D 20 69 73 20 66 69 78 65 64 20 21)

REPORT(Testing for DM1, DH1, DM3, DH3, DM5, DH5 packets)
report( )

TXDATA(hCon:0,bc:0,pb:2,Len:1,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:2,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:3,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:4,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:5,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:6,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:7,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:8,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:9,cnt:10,Random:0)
TXDATA(hCon:0,bc:0,pb:2,Len:10,cnt:10,Random:0)

report()
report(Packets size = 1..10 ”passed”)
report()

// Disconnect ACL connection

TXCMD 06 04 03 00 00 13
WAITEVENT($05,10000,[TestError])
report(ACL connection disconnected)
report( )

label: TestSuccess
report(Test passed!)
report( )
jump(end)

label: TestError
report(Test failed!)
label: end
REPORT(DONE!)
```

Appendices

Appendix A: Regulatory Statements

This product complies with any mandatory product specification in any country where the product is sold. Additionally, the product complies with the following:

United States of America and Canada

Tested to comply with FCC Standard FOR HOME OR OFFICE USE. See FCC 47CFR, part 15.19(b)(2).

This device complies with part 15 of the FCC rules and with RSS-210/RSS-139 of the Industry Canada. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note that any changes or modifications to this equipment not expressly approved by the manufacturer may void the FCC authorization to operate this equipment.

European Union (EU) and EFTA

This equipment complies with the R&TTE directive and has been provided with the CE mark accordingly.

Note that the radio frequency band used by this equipment has not been harmonized in all of the EU.

Glossary

Glossary

ACL

An acronym for Asynchronous Connection-Less link, this provides a packet-switched connection (master to any slave).

Active Member Address (AM_ADDR)

The Active Member Address is a 3-bit number. This address is allocated by the master to each active slave in the piconet. The address is used to identify the specific slave for which a packet is intended.

Authentication

Security mechanism that prevents access to critical data and makes it impossible to falsify the origin of a message. Authentication is performed for devices. In Bluetooth, this is achieved by the authentication procedure based on the stored link key or by pairing (entering a PIN).

AUX

An ACL (asynchronous connectionless) link packet type for data. An AUX1 packet resembles a DH1 packet except it has no CRC code. As a result it can carry up to 30 information bytes.

Baseband

The baseband describes the specifications of the Bluetooth link controller, which carries out the baseband protocols and other low-level link routines.

BD_ADDR

The Bluetooth Device Address is a unique, 48-bit number used to identify a Bluetooth device. The Bluetooth device address is also used in encryption and in generation of frequency hop sequences. It is similar to an Ethernet MAC address.

Bluetooth

An open specification for wireless communication of data and voice. It is based on a low-cost, short-range radio link facilitating protected ad hoc connections for stationary and mobile communication environments.

Bluetooth Clock

Every Bluetooth unit has an internal system clock which determines the timing and hopping of the transceiver. It can be implemented as a 28-bit counter, with the LSB ticking in units of 312.5us, giving a clock rate of 3.2kHz.

Bluetooth Device Class

A parameter that indicates the type of device and which types of services that are supported. The class is received during the discovery procedure.

Bluetooth Host

This is a computing device, peripheral, cellular telephone, access point to PSTN (public switched telephone network), etc. This host attached to a Bluetooth unit may communicate with other Bluetooth hosts attached to their Bluetooth units as well.

Bluetooth Neighborhood

A Bluetooth application created by Digianswer that provides an interface for you to interact with Bluetooth systems. Its basic functions are to perform device and service discovery and to enable you to make service-oriented connections to other Bluetooth devices.

Bluetooth Service Type

One or more services a device can provide to other devices. The service information is defined in the service class field of the Bluetooth device class parameter.

Bluetooth Unit

A voice/data circuit equipment for a short-range, wireless communication link. It allows voice and data communications between Bluetooth units.

Channel

A logical connection on the L2CAP level between two devices serving a single application or higher layer protocol.

Channel (Hopping) Sequence

This is a pseudo-random sequence of 79 (23 for the 23MHz system) frequencies. The frequency is calculated using the BD_ADDR of the master of the piconet. The phase in the sequence is derived from an estimate of the master clock. The channel hopping sequence has a very long period length that does not show repetitive patterns over a short time interval, but which distributes the hop frequencies equally over the 79 MHz (23 MHz for the 23 MHz system) during a short time interval. See also Frequency sequence.

CID (Channel Identifier)

An abbreviation for Channel Identifier. Used to identify L2CAP connections.

CLK

An acronym for Clock, this is the master clock that defines the timing used on a Bluetooth piconet.

CLKE

An estimate of the clock of another device.

CLKN

The native clock of a Bluetooth device. A slave device must add an offset to its own CLKN to synchronize with the master clock (CLK).

Coverage Area

The area where two Bluetooth units can exchange messages with acceptable quality and performance.

Destination

The Bluetooth device receiving an action from another Bluetooth device. The device sending the action is called the source. The destination is typically part of an established link, though not always (such as in inquiry/page procedures).

Device Discovery

Before a link can be established, a Bluetooth device needs to discover the other Bluetooth devices that are active within the range. The mechanism to request and receive the Bluetooth address, clock, class of device, used page scan, and names of devices is referred to as device discovery.

Device Name

The name that a Bluetooth device presents when supplying identity information to another device.

DH (Data-High Rate)

An ACL link data packet type for high rate data. DH1 packets are similar to DM1 packets, except that the information in the payload is not FEC encoded. This means the DH1 packet can carry up to 28 information bytes and covers a single time slot. The DH3 is the same except it can cover up to 3 time slots and contain up to 185 information bytes. The DH5 packet is the same again except it can cover up to 5 time slots and contains up to 341 information bytes.

Discoverable Device

A Bluetooth device in range that will respond to an inquiry message.

DM (Data-Medium Rate)

An ACL link data packet type for medium rate data. DM1 packets carry information data only, containing a 16-bit CRC code and up to 18 info bytes. They are encoded using 2/3 FEC and the packet can cover up to a single time slot. DM3 packets are the same except they can cover up to 3 time slots, and can carry up to 123 information bytes. DM5 packets are the same again except they can cover up to 5 time slots and can hold up to 226 information bytes.

DV (Data Voice)

A SCO (synchronous connection oriented) link data packet type for data and voice. It is divided into a voice field of 80 bits and a data field of 150 bits. The voice field is not covered by FEC, but the data field is covered by 2/3 FEC. The voice and data fields are treated completely separate. The voice field is handled like normal SCO data and is never retransmitted; that is, the voice field is always new. The data field is checked for errors and is retransmitted, if necessary.

Encryption

Security mechanism that prevents eavesdropping and maintains link privacy.

FEC (Forward Error Correction)

The purpose of the FEC scheme on the data payload is to reduce the number of retransmissions. Within Bluetooth, there are 2 versions of this: 1/3 FEC and 2/3 FEC. 1/3 FEC is a simple, 3-times repetition of each information bit. 2/3 FEC is a (15,10) shortened Hamming code.

Frequency Hopping (Selection)

Bluetooth is characterized by its system of fast frequency hops. 10 different types of hopping sequences are defined, 5 of the 79 MHz range/79 hop system and 5 for the 23 MHz range/23 hop system. The two range system hopping sequences differ only in frequency range 79 MHz or 23 MHz, and segment length: 32 hops (79 MHz system) or 16 hops (23 MHz system).

The individual hopping sequences include the page sequence and the page response sequence. These are used in the page procedure. Used in the inquiry procedure are the inquiry sequence and the inquiry response sequence. Finally the main hopping sequence used in the Bluetooth system is the channel hopping sequence.

Frequency Hopping Synchronization (FHS) Packet

This a special control packet revealing, among other things, the BD_ADDR and the clock of the source device. It contains 144 information bits and a 16-bit CRC code. The payload is coded with a rate 2/3 FEC, which brings the total payload length to 240 bits. The FHS packet covers a single time slot.

Gateway

A Bluetooth enabled base station that is connected to an external network.

Hold Mode

Devices synchronized to a piconet can enter power-saving modes in which device activity is lowered. The master unit can put slave units into HOLD mode, where only an internal timer is running. Slave units can also demand to be put into HOLD mode. Data transfer restarts instantly when units transition out of HOLD mode. It has an intermediate duty cycle (medium power efficient) for the 3 power saving modes (sniff, hold, and park).

Host Controller Interface (HCI)

Allows higher layers of the stack, including applications to access the baseband, link manager, and other hardware registers through a single, standard interface.

HV (High Quality Voice)

A SCO link voice packet. HV1 packets carry 10 information bytes, which are protected by 1/3 FEC. HV2 packets carry 20 information bytes and are protected by 2/3 FEC. HV3 packets carry 30 information bytes and not protected by FEC. HV packets do not have a CRC or payload header.

Inquiry

A Bluetooth unit transmits inquiry messages to discover the other Bluetooth units active within the coverage area. Units that capture inquiry messages may send a response to the inquiring Bluetooth unit. The response contains information about the Bluetooth unit and its inquiring host.

Isochronous User Channel

A channel used for time bounded information such as compressed audio (ACL link).

L2CAP

Acronym for Logical Link Controller and Adaptation Protocol.

LAN

Acronym for Local Area Network.

LMP

Acronym for Link Manager Protocol. The LMP is used for link setup and control. The LMP PDU signals are interpreted and filtered out by the Link Manager on the receiving side and are not propagated to higher layers.

Logical Channel

The different types of channels on a physical link.

Master Device

The device that initiates a connection and, during this connection, controls all traffic in a piconet. The clock and hopping sequence of the master are used to synchronize all other devices in the piconet.

Name Discovery

The mechanism to request and receive a device name.

OBEX

An abbreviation for OBject EXchange protocol. The OBEX tab displays file-transfer and business card data.

NULL packet

A 126-bit packet consisting of the CAC (channel access code) and packet header only. It is used to return link information to the source. The NULL packet does not have to be acknowledged.

Packet

Format of aggregated bits that can be transmitted in 1, 3, or 5 time slots.

Paging

A Bluetooth unit transmits paging messages to set up a communication link to another Bluetooth unit that is active within the coverage area.

Park Mode

In the PARK mode, a device is still synchronized to the piconet but does not participate in the traffic. Parked devices have given up their MAC (AM_ADDR) address and occasionally listen to the traffic of the master to resynchronize and check on broadcast messages. This mode has the lowest duty cycle (power efficiency) of the three power-saving modes (sniff, hold, and park).

PDU

Acronym for Protocol Data Unit (that is, a message).

Physical Channel

Synchronized RF hopping in a piconet.

Physical Link

Connection between devices.

Piconet

A wireless network formed by two or more Bluetooth devices.

POLL Packet

Similar to the NULL packet, except it requires a confirmation from the destination. Upon reception of a POLL packet, the slave must respond with a packet.

Profile

Application that a Bluetooth device facilitates. For one device to communicate with another, the two devices must have a shared profile. For example, to transfer files from one computer to another, both computers must feature the file transfer profile.

Protocol Stack

Allows device to locate, connect to, and exchange data with each other and to execute interoperable, interactive applications against each other. The stack is logically partitioned into three groups: transport protocol, middleware protocol, and application group.

RFCOMM

Serial Cable Emulation Protocol based on ETSI TS 07.10. (European Telecommunications Standards Institute).

RX

Abbreviation for receive.

Scatternet

Multiple independent and nonsynchronized piconets form a scatternet.

SDP (Service Discovery Protocol)

SDP is a Bluetooth-defined protocol provided for or available through a Bluetooth device. This protocol essentially is a means for applications to discover which services are available and to determine the characteristics of those available services.

Slave

A device in a piconet controlled by another device (the master).

Sniff Mode

Devices synchronized to a piconet can enter power-saving modes in which device activity is lowered. In the SNIFF mode, a slave device listens to the piconet at reduced rate, thus reducing the duty cycle. The SNIFF interval is programmable and depends on the application. It has the highest duty cycle (least power efficient) of all 3 power saving modes (sniff, hold and park).

Source

The Bluetooth device initiating an action to another Bluetooth device. The device receiving the action is called the destination. The source is typically part of an established link, although not always (such as in inquiry/page procedures).

Time Slot

A time slot is the time it takes to send one packet from one Bluetooth device to another. A single time slot in a Bluetooth system lasts 625 us.

TCS

Acronym for Telephony Control (protocol) Specification. The TCS tab displays protocol discriminator, message type, and other data (depending upon the message type).

TX

Abbreviation for transmit.

Index

Index